GRASP CONTACT BETWEEN HAND AND OBJECT: CAPTURE, ANALYSIS, AND APPLICATIONS

A Dissertation
Presented to
The Academic Faculty

By

Samarth Brahmbhatt

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Interactive Computing, College of Computing

Georgia Institute of Technology

August 2020

GRASP CONTACT BETWEEN HAND AND OBJECT: CAPTURE, ANALYSIS, AND APPLICATIONS

Approved by:

Dr. James Hays, Advisor
School of Interactive Computing
Georgia Institute of Technology

Dr. Charles C. Kemp
Department of Biomedical Engineering
Georgia Institute of Technology

Dr. James M. Rehg
School of Interactive Computing
Georgia Institute of Technology

Dr. C. Karen Liu
Computer Science Department
Stanford University

Dr. Yaser Ajmal Sheikh
Robotics Institute
Carnegie Mellon University

Date Approved: April 30, 2020

कर्मण्येवाधिकारस्ते मा फलेषु कदाचन ।

मा कर्मफलहेतुर्भुर्मा ते संगोऽस्त्वकर्मणि ॥

(You have the right to work only, but never to its fruits. Let not the fruits of action be your motive, nor let your attachment be to inaction.)

The Bhagavad Gita, Chapter 2, Verse 47

To Sunny Kaka

ACKNOWLEDGEMENTS

This thesis would not have been possible without James, my PhD advisor. He has not only mentored my research very well technically, but also given me space and responsibilities that encouraged me to be independent. Charlie's enthusiasm and technical advice have been equally important. His lab focused on robots doing good for society is an inspiration for me.

I also acknowledge the love and support of my family, especially my parents, sister Siddhi, and wife Emily. They have been instrumental in my PhD journey in ways too many to be enumerated.

Thank you also to labmates in James' group – Amit, Cusuh, Sean, John, and Patsorn, in Charlie's group – Henry, Tapo, Ari, Zackory and Patrick. Thanks are also due to Siddhartha, Zhaoyang, Huda, and of course, Abhay and Rakshit. I would also like to thank the good folks at FRL Nimble – especially Chris, Chengcheng, and Rob – for being so supportive of my research.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Contact is an important but often oversimplified component of human grasping. Capturing hand-object contact in detail can lead to important insights about grasping behavior, and enable applications in diverse fields like virtual reality and human-robot interaction. However, observing contact through external sensors is challenging because of occlusion and the complexity of the human hand. Lack of ground-truth data has significantly influenced research in this field. This thesis introduces the use of thermal cameras to capture detailed ground-truth hand-object contact (called contact maps), and techniques to simultaneously capture other data modalities like 3D hand pose, object pose, and multi-view RGB-D grasp videos. This has resulted in ContactDB and ContactPose, two large-scale and diverse datasets of participants grasping 3D-printed household objects with functional intents. Analysis of this data confirms some long held intuitions about hand-object contact, and also reveals some surprising new patterns. We also train machine learning models for diverse contact map prediction from object shape, and for contact modeling from object shape and grasp information. Next, this thesis presents ContactGrasp, an algorithm that uses object shape and a contact map to synthesize functional grasps for kinematically diverse hand models, including robotic end-effectors. Finally, this thesis investigates whether the contact data captured by thermal cameras encodes contact pressure in addition to contact locations. We find that (subject to certain conditions) the structure of our contact data indeed includes information about contact pressure.

CHAPTER 1
INTRODUCTION

A significant portion of human physical activity involves grasping and manipulation of objects. Consequently, we have designed objects and surfaces around us to match them to our cognitive and physical abilities. It has been a scientific dream since centuries to make machines with similar abilities, so that they can work efficiently – not in artificially structured environments, but in environments already familiar to us – and ultimately help us. A big unsolved problem in this grand goal is grasping. How can a machine (robot) figure out which tools to use for a task, find where they are, pick them up, hold them, and use them? This thesis focuses on the 'hold them' and 'use them' parts.

The challenging nature of the problem and the proliferation of freely available human activity data in the form of images and videos has increased the attractiveness of the immitation approach to solution. The immitation approach seeks to obseve how humans do it, and learn. In contrast, the 'search' approach tries to search for the solution by trial and error (mostly in a simulator, but sometimes in the real world) and focuses on intelligently picking solutions to try. This thesis develops new tools to observe aspects of human grasping behavior which were previously challenging to observe with precision, in the hope that this can lead to better immitation.

We introduce the use of *thermal cameras* to observe hand-object contact in high detail. Contact is obviously a very important part of grasping and manipulation, and therefore has been reasoned about mathematically extensively. Our newfound ability to *observe* it has confirmed some long held intuitions about it, and also revealed some surprising new patterns. Our method allows for high-quality capture of contact locations on the object surface for static grasps. We have used it to create ContactDB,

the first large-scale and diverse dataset in this field. Chapter 2 discusses more details, including motivations for contact capture.

Given this ability to capture hand-object contact, we were next interested in capturing commonly used data modalities like images and hand-object poses along with contact. This is essential for learning models that predict contact (a useful but not easily observable quantity) from hand pose and images (quantities that can be easily captured with today's sensors and algorithms). This resulted in a new dataset, ContactPose, that captures multi-view RGB-D grasp videos in addition to hand-object contact, and annotates each frame in the videos with hand and object pose, largely automatically. Chapter 3 discusses more details.

Demonstrations from humans for learning to immitate their grasping behavior have traditionally been collected in the human hand configuration space, by making participants wear gloves or other tracking devices. This makes them difficult to use with robot hands that have a different parameterization or geometry, and can also interfere with natural grasping behavior. Through the ContactGrasp algorithm, we show how ContactDB and ContactPose data can be used as grasping demonstrations, in the hand-object contact space. Chapter 4 discusses how and when this is desirable, along with implementation details.

The final experiment described in this thesis (Chapter 5) investigates whether the contact data captured by thermal cameras (described in Chapters 2 and 3) encodes contact pressure in addition to contact locations. If true, this can further complete the description of human grasping behavior captured by our proposed methods, and enable better immitation. We find that (subject to certain conditions) the structure of our contact data indeed includes information about contact pressure. Chapter 5 discusses details, including proposals for relaxing these conditions.

To summarize, this thesis investigates the capture of hand-object contact while humans are grasping household objects. It presents analysis of this data, describes

how models to predict contact from various data modalities can be trained, and proposes the use of contact data as demonstrations of human grasping behavior for robotic grasping.

CHAPTER 2

CONTACTDB: ANALYZING AND PREDICTING GRASP CONTACT VIA THERMAL IMAGING

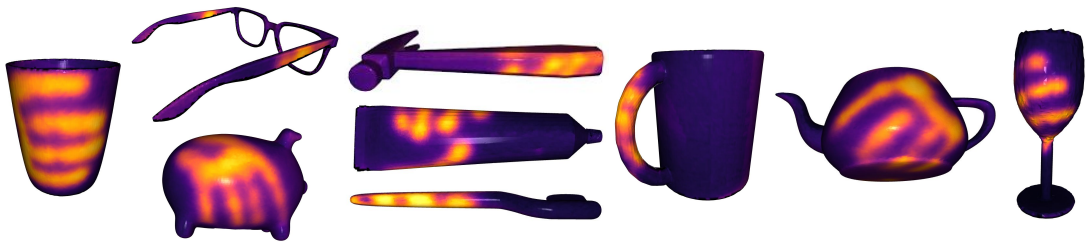


Figure 2.1: Example contact maps from ContactDB, constructed from multiple 2D thermal images of hand-object contact resulting from human grasps.

Abstract: Grasping and manipulating objects is an important human skill. Since hand-object contact is fundamental to grasping, capturing it can lead to important insights. However, observing contact through external sensors is challenging because of occlusion and the complexity of the human hand. We present ContactDB, a novel dataset of contact maps for household objects that captures the rich hand-object contact that occurs during grasping, enabled by use of a thermal camera. Participants in our study grasped 3D printed objects with a post-grasp functional intent. ContactDB includes 3750 3D meshes of 50 household objects textured with contact maps and 375K frames of synchronized RGB-D+thermal images. To the best of our knowledge, this is the first large-scale dataset that records detailed contact maps for human grasps. Analysis of this data shows the influence of functional intent and object size on grasping, the tendency to touch/avoid 'active areas', and the high frequency of palm and proximal finger contact. Finally, we train state-of-the-art image translation and 3D convolution algorithms to predict diverse contact patterns from object shape. Data, code and models are available at `https://contactdb.cc.gatech.edu`.

2.1 Introduction

Humans excel at grasping and then performing tasks with household objects. Human grasps exhibit contact locations, forces and stability that allows post-grasp actions with objects, and are also significantly influenced by the post-grasp intent [1, 2, 3]. For example, people typically grasp a knife by the handle to use it, but grasp it by the blunt side of the blade to hand it off.

A large body of previous work [4, 5, 6, 7, 8, 9, 10, 11, 12, 6, 12, 13, 7] has recorded human grasps, with methods ranging from data gloves that measure joint configuration to manually arranged robotic hands. ContactDB differs significantly from these previous datasets by *focusing primarily on the contact* resulting from the rich interaction between hand and object. Specifically, we represent contact through the texture of 3D object meshes, which we call 'contact maps' (see Figure 2.1).

There are multiple motivations for recording grasping activity through contact maps. Since it is *object-centric*, it enables detailed analysis of grasping preferences influenced by functional intent, object shape, size and semantic category, and learning object shape features for grasp prediction, and grasp re-targeting to kinematically diverse hand models. Previously employed methods of recording grasping activity do not easily support such analysis, as we discuss in Section 2.2.

We created ContactDB by recording human participants grasping a set of 3D printed household objects in our laboratory, with two different post-grasp functional intents–using the object and handing it off. See Section 2.3 for more details on the data collection procedure, size of the dataset and the kinds of data included.

Except for contact edges viewed from select angles, and contact with transparent objects, contact regions are typically occluded from visual light imaging. Hence, existing studies on the capture and analysis of hand-object contact are extremely limited. Fundamental questions such as the role of the palm in grasping everyday

objects are unanswered. We propose a novel procedure to capture contact maps on the object surface at unprecedented detail using an RGB-D + thermal camera calibrated rig.

We make the following contributions:

- **Dataset**: Present a dataset recording functional human grasping consisting of 3750 meshes textured with contact maps and 375K frames of paired RGBD-thermal data.
- **Analysis**: Demonstrate the influence of object shape, size and functional intent on grasps, and show the importance of non-fingertip contact.
- **Prediction**: Explore data representations and diverse prediction algorithms to predict contact maps from object shape.

2.2 Related Work

2.2.1 Datasets of Human Grasps

Since contact between the human hand and an object is fundamental to grasping and manipulation, capturing this contact can potentially lead to important insights about human grasping and manipulation. In practice, however, this has been a challenging goal. The human hand is highly complex with extensive soft tissue and a skeletal structure that is often modeled with 26 degrees of freedom. Hence, previous work has focused on recording grasping activity in other forms like hand joint configuration by manual annotation [8, 9], data gloves [4, 5] or wired magnetic trackers [14, 15] (which can interfere with natural grasping), or model-based hand pose estimation [10]. At a higher level, grasping has been observed through third-person [11, 12, 6] or first-person [12, 13, 7] videos, in which frames are annotated with the category of grasp according to a grasp taxonomy [16, 17]. Tactile sensors are embedded on a glove [18] or in the object [19] to record grasp contact points. Such methods are limited by the

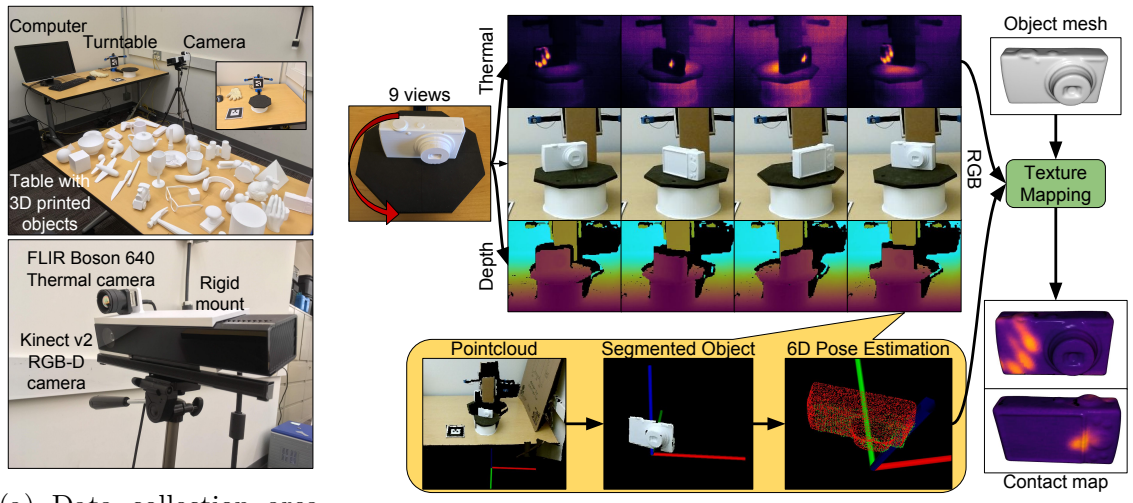
resolution of tactile sensors. Puhlmann et al [20] capture hand-table contact during grasping with a touchscreen. Rogez et al [21] manually configure a hand model to match grasps from a taxonomy, and use connected component analysis on hand vertices intersecting with an object model to estimate contact regions on the hand.

Due to hand complexity and lack of understanding of how humans control their hands, approaches like those mentioned above have so far been limited to providing coarse or speculative contact estimates. In contrast, our approach allows us to directly observe where contact between the object and the human hand has taken place with an unprecedented level of fidelity.

2.2.2 Predicting Grasp Contact

Our work is related to that of Lau et al [22], which crowdsources grasp tactile saliency. Online annotators are instructed to choose a point they would prefer to touch, from a pair sampled from the object surface. This pairwise information is integrated to construct the tactile saliency map. In contrast, ContactDB contact maps are full observations of real human grasps with functional intent (see Appendix A for a qualitative comparison). Akizuki et al [23] use hand pose estimation and model-based object tracking in RGB-D videos to record a set of contact points on the object surface. This is vulnerable to inaccuracies in the hand model and hand pose tracking. Hamer at al [24] record human demonstrations of grasping by registering depth images to get object geometry and object- and hand-pose. Contact is approximated as a single point per fingertip. A large body of work in robotics aims to predict a configuration of the end-effector [25, 26, 27] suitable for grasping. In contrast to ContactDB, these works model contact as a single point per hand digit, ignoring other contact.

Diverse Predictions: Grasping is a task where multiple predictions can be equally correct. Lee et al [28] and Firman et al [29] have developed theoretical frameworks allowing neural networks to make diverse and meaningful predictions. Recently,



(a) Data collection area setup

(b) Data processing pipeline, explained in detail in Section 2.3.3

Figure 2.2: Data collection and processing for ContactDB. Participants grasp 3D printed objects and put them on the rotating turntable. Thermal images from multiple views are texture-mapped to the object mesh.

Ghazaei et al [30] have used similar techniques to predict diverse grasp configurations for a parallel jaw gripper.

2.3 The ContactDB Dataset

Here we present the design choices and process in creating the ContactDB, which consists of 50 3D printed household objects being grasped with two functional intents by 50 participants (see Table 2.1).

Observing Contact Through a Thermal Camera. At the core of our data collection process is the use of a thermal camera to observe the precise locations of contact between human hand and object. Thermal cameras have recently been used to capture humans and their interaction with the environment. For example, Luo et al [31] observe humans interacting with objects for egocentric SLAM, while Larson et al [32] observe human finger interaction with arbitrary surfaces to make them interactive. Both note the phenomenon of thermally observable contact, but do not investigate it rigorously or collect a large-scale dataset.

Table 2.1: Size of the ContactDB Dataset

	Functional Intent		**Total**
	Use	**Hand-off**	
Participants	50	50 (same)	
Objects	27	48 (overlapping)	50
Textured meshes	1350	2400	**3750**
RGBD-Thermal frames	135K	240K	**375K**

When a participant grasps an object, heat from the hand transfers onto the object surface. If the object material does not dissipate the heat rapidly, the precise contact areas can be clearly observed in the thermal image after the object is released (see Figure 2.2b). Intensity at a pixel in the thermal image is a function of the infrared energy emitted by the corresponding world point [33]. Hence, object pixel intensity in our thermal images is related to heat of the skin, duration of contact, heat conduction (including diffusion to nearby object locations), and contact pressure. By keeping these factors roughly constant during data collection, we verified empirically that heat conduction from hand-object contact is the dominant factor in the observed thermal measurements. See Appendix A for more discussion on heat dissipation and accuracy.

2.3.1 Object Selection and Fabrication

We decided to focus on household objects since an understanding of contact preferences and the ability to predict them are most likely to improve human-robot interaction in household settings. Other standard grasping datasets [34] and competitions [35] have a similar focus. We started with the YCB dataset [34] to choose the 50 objects in our dataset. We excluded similarly-shaped objects (e.g. cereal and cracker boxes) that are unlikely to produce different kinds of grasps, deformable objects (e.g. sponge, plastic chain, nylon rope), very small (e.g. dominoes, washers), and very large objects (e.g. cooking skillet, Windex bottle). We added common ones such as flashlight, eyeglasses, computer mouse, and objects popular in computer

graphics (e.g. Stanford bunny and Utah teapot). Since object size has been shown to influence the grasp [36, 1] and we are interested in contact during grasping of abstract shapes, we included 5 primitive objects–cube, cylinder, pyramid, torus and sphere–at 3 different scales (principal axes 12, 8 and 4 cm). See Appendix A for a full object list.

We chose to 3D print all the objects to ensure uniform heat dissipation properties. Additionally, we empirically found that the PLA material used for 3D printing is excellent for retaining thermal handprints. We used open-source resources to select suitable models for each object, and printed them at 15% infill density using white PLA filament on a Dremel 3D20 printer. 3D printing the objects has additional advantages. Having an accurate 3D model of the object makes 6D pose estimation of the object from recorded pointcloud data easier (see Section 2.3.3), which we use for texture mapping contact maps to the object mesh. 3D printing the objects also allows participants to focus on the object geometry during grasping.

2.3.2 Data Collection Protocol

Figure 2.2a shows our setup. We rigidly mounted a FLIR Boson 640 thermal camera on a Kinect v2 RGB-D sensor. The instrinsics of both the cameras and extrinsics between them are calibrated using ROS [37], so that both RGB and depth images from the Kinect can be accurately registered to the thermal image. We invited 50 participants (mostly 20-25 years of age, able-bodied males and females), and used the following protocol approved by the Georgia Tech Institutional Review Board.

50 3D printed objects were placed at random locations on a table in orientations commonly encountered in practice. Participants were asked to grasp each object with a post-grasp functional intent. They held the object for 5 seconds to allow heat transfer from the hand to the object, and then hand it to an experimenter. The experimenter wore an insulating glove to prevent heat transfer from their hand, and

places the object on a turntable about 1 m away from the cameras. Participants were provided with chemical hand warmers to increase the intensity of thermal handprints. The cameras recorded a continuous stream of RGB, depth and thermal images as the turntable rotated in a 360 degree arc. The turntable paused at 9 equally spaced locations on this arc, where the rotation angle of the turntable was also recorded. In some cases, objects were flipped and scanned a second time to capture any thermal prints that were unseen in the previous rotation.

We used two post-grasp *functional intents*: 'use' and 'hand-off'. Participants were instructed to grasp 48 objects with the intent of handing them off to the experimenter, and to grasp a subset of 27 objects (after the previous thermal handprints had dissipated) with the intent of using them. We used only a subset of 27 objects for 'use', since other objects (e.g. pyramid, Stanford bunny) lack clear use cases. See the Appendix A for specific use instructions. Participants were asked to avoid in-hand manipulation after grasping to avoid smudging the thermal handprints.

2.3.3 Data Processing

As the turntable rotates with the object on it, the stream of RGB-D and thermal images capture the object from multiple viewpoints. The aim of data processing is to texture-map the thermal images to the object 3D mesh and generate a coherent contact map (examples are shown in Figure 2.1).

The entire process is shown in Figure 2.2b. We first extracted the corresponding turntable angle and RGB, depth and thermal images at the 9 locations where the turntable pauses. Next, we converted the depth maps to pointclouds and useed a least-squares estimate of the turntable plane and white color segmentation to segment the object. We used the Iterative Closest Point (ICP) [38] algorithm implemented in PCL [39] to estimate the full 6D pose of the object in the 9 segmented pointclouds. Object origins in the 9 views were used to get a least squares estimate of the 3D circle

described by the moving object. This circle was used to interpolate the object poses for views which are unsuitable for the ICP step because of noise in the depth map or important shape elements of the object being hidden in that view, or for rotating symmetric objects around the axis of symmetry.

Finally, the 3D mesh along with the 9 pose estimates and thermal images were input to the colormap optimization algorithm of [40], which is implemented in Open3D [41]. It locally optimizes object poses to minimize the photometric texture projection error and generates a mesh coherently textured with contact maps.

2.4 Analysis of Contact Maps

In this section we present analysis of some aspects of human grasping, using the data in ContactDB. We processed each contact map separately to increase contrast by applying a sigmoid function to the texture-mapped intensity values that maps the minimum to 0.05 and maximum to 0.95.

Effect of Functional Intent. We observed that the functional intent ('use' or 'hand off') significantly influences the contact patterns for many objects. To show qualitative examples, we clustered the contact maps within each object and functional intent category using k-medoids clustering [42] ($k = 3$) on the XYZ values of points which have contact value above 0.4. The distance function between two sets of points was defined as $d(\mathbf{p}_1, \mathbf{p}_2) = \left(\bar{d}(\mathbf{p}_1, \mathbf{p}_2) + \bar{d}(\mathbf{p}_2, \mathbf{p}_1)\right) / \left(|\mathbf{p}_1| + |\mathbf{p}_2|\right)$, where $\bar{d}(\mathbf{p}_1, \mathbf{p}_2) = \sum_{i=1}^{|\mathbf{p}_1|} \min_{j=1}^{|\mathbf{p}_2|} ||\mathbf{p}_1^{(i)} - \mathbf{p}_2^{(j)}||_2$. For symmetric objects, we chose the angle of rotation around the axis of symmetry that minimized $d(\mathbf{p}_1, \mathbf{p}_2)$. Figure 2.3 shows dominant contact maps (center of the largest cluster) for the two different functional intents.

To quantify the influence of functional intent, we define 'active areas' (highlighted in green in Figure 2.3) on the surface of some objects and show the fraction of participants that touched that area (evidenced by the map value being greater than 0.4) in Table 2.2.

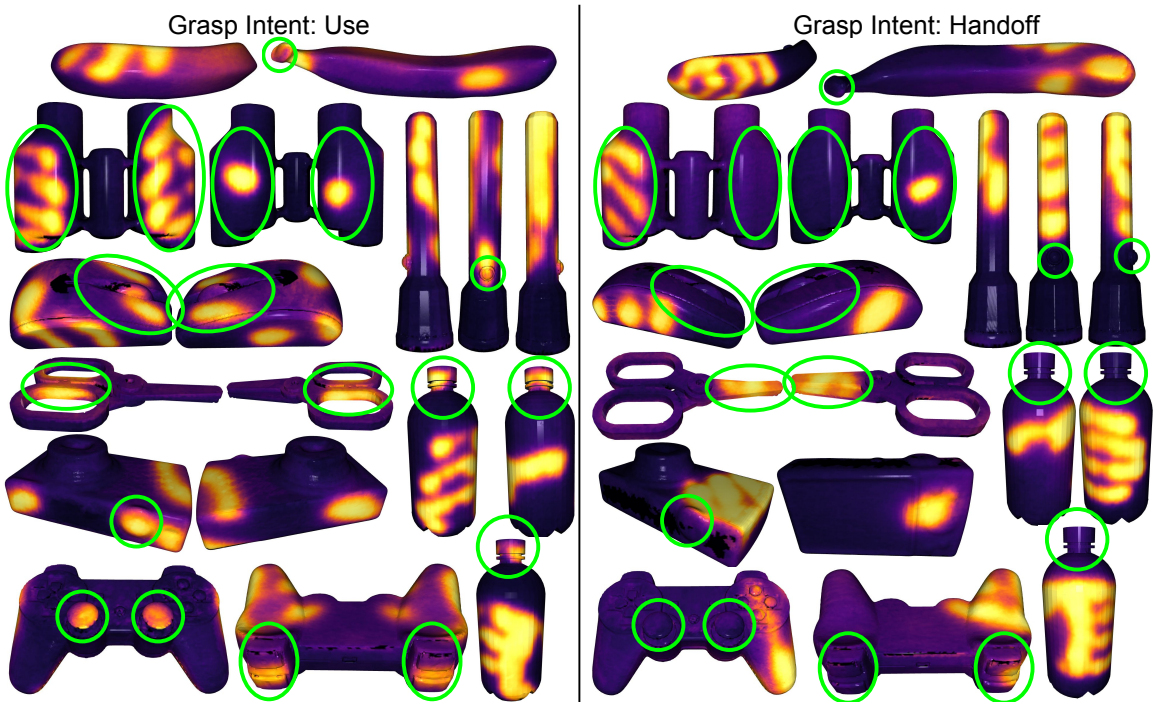


Figure 2.3: Influence of functional intent on contact: Two views of the dominant grasp (center of the largest cluster after k-medoids clustering across participants). Green circles indicate 'active areas'. This influence is quantified in Table 2.2.

Table 2.2: Fraction of participants that touched active areas for different functional intents. See Fig. 2.3 for examples.

Active Area	**handoff**	**use**
Banana tip (either tip)	22.45	63.27
Binoculars (both barrels)	12.50	93.88
Camera shutter button	34.00	69.39
Eyeglasses (both temples)	4.00	64.58
Flashlight button	28.00	62.00
Hammer (head)	38.00	0.00
Mouse (both click buttons)	16.00	84.00
PS controller (both front buttons)	2.00	40.81
PS controller (both analog sticks)	2.00	22.44
Scissors (handle)	38.00	100.00
Scissors (blade)	60.00	0.00
Water-bottle cap	16.00	67.35
Wine glass stem	56.00	30.61

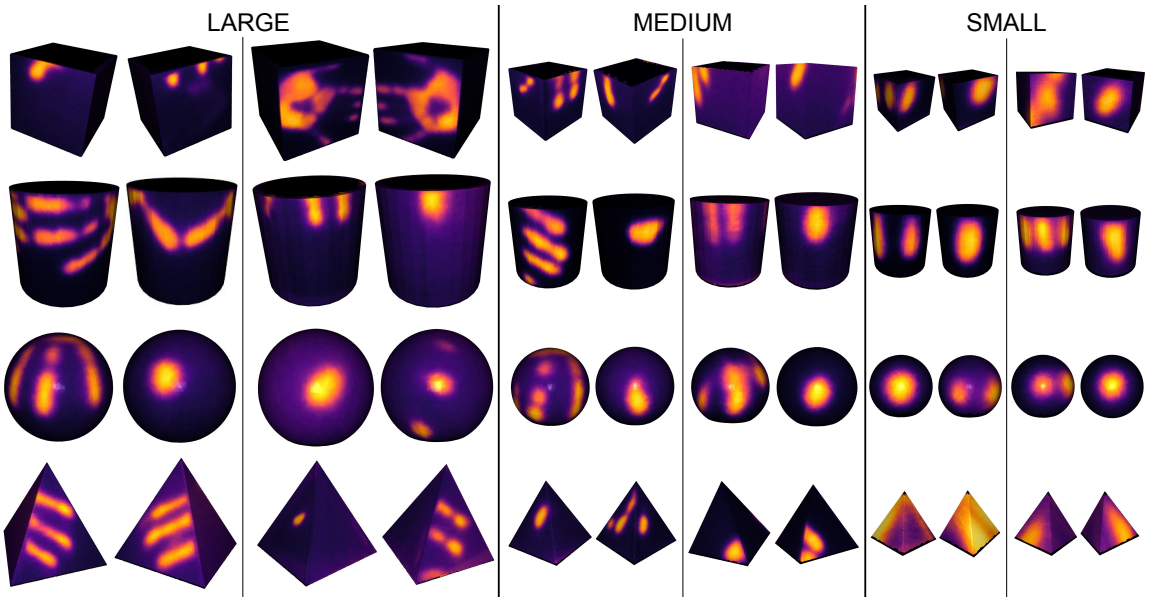


Figure 2.4: Influence of object size on contact: Two dominant grasps for objects of same shape and varying size.

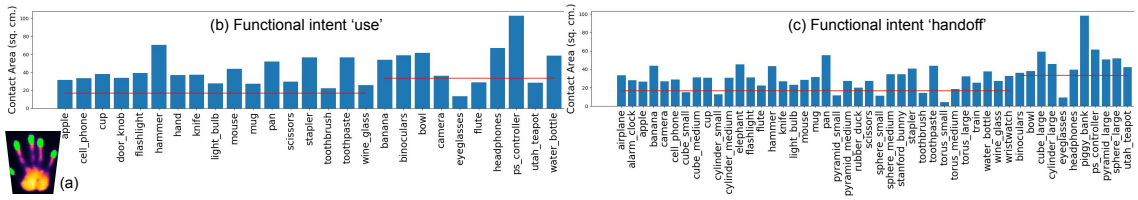


Figure 2.5: (a): Palm contact on plate, annotated fingertips. (b, c): Contact areas for objects in ContactDB, averaged across participants. The red line indicates a loose upper bound on contact area for a fingertip-only grasp, which is doubled for objects which have bimanual grasps.

Effect of object size. Figure 2.4 shows the dominant contact maps for objects of the same shape at three different sizes. Small objects exhibit grasps with two or three fingertips, while larger objects are often grasped with more fingers and more than the fingertips in contact with the object. Grasps for large objects are bi-modal: bimanual using the full hands, or single-handed using fingertips. To quantify this, we manually labelled grasps as bimanual/single-handed, and show their relation to hand size in Fig. 2.6. The figure shows that people with smaller hands prefer to grasp `large` objects (for 'handoff') with bimanual grasps. No bimanual grasps were observed for the `medium` and `small` object sizes.

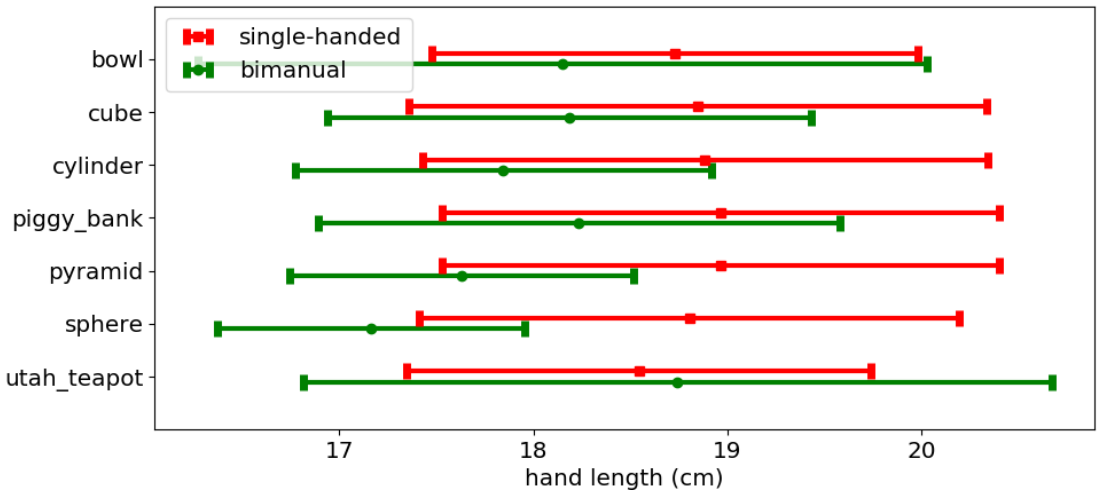


Figure 2.6: Relationship between hand length (wrist to mid fingertip) and single-handed/bimanual grasps. The intervals show mean and 1 standard deviation. Cube, cylinder, pyramid and sphere are of the `large` size.

How much of the contact is fingertips? Contact is traditionally modelled in robotics [43] and simulation [44] as a single point. However, the contact maps in Figures 2.1, 2.3 and 2.4 show that human grasps have much more than fingertip contact. Single-point contact modeling is inspired by the prevalence of rigid manipulators on robots, but with the recent research interest in *soft robots* [45, 46], we now have access to manipulators that contact the object at other areas on the finger. Data in ContactDB shows the use of non-fingertip contact for highly capable soft manipulators: human hands. For each contact map, we calculated the contact area by integrating the area of all the contacted faces in the mesh. A face is contacted if any of its three vertices have a contact value greater than 0.4. Figures 2.5(b) and 2.5(c) show the contact areas for all objects under both functional intents, averaged across participants. Next, we calculated an upper bound on the contact area if only all 5 fingertips were touching the object. This was done by capturing the participants' palm print on a flat plate, where it is easy to manually annotate the fingertip regions (shown in Figure 2.5(a)). The total surface area of fingertips in the palm print is the desired upper bound. It was doubled for objects for which we observe bimanual

grasps. This upper bound was averaged across four participants, and is shown as the red line in Figures 2.5(b) and 2.5(c). Note that this is a loose upper bound, since many real-world fingertip-only grasps don't involve all five fingertips, and we mark the entire object category as bimanual if even one participant performs a bimanual grasp. Total contact area for many objects is significantly higher than the upper bound on fingertip-only contact area, indicating the large role that the soft tissue of the human hand plays in grasping and manipulation. This motivates the inclusion of non-fingertip areas in grasp prediction and modeling algorithms, and presents an opportunity to inform the design of soft robotic manipulators. Interestingly, the average contact area for some objects (e.g. bowl, mug, PS controller, toothbrush) differs across functional intent, due to different kinds of grasps used.

2.5 Predicting Contact Maps

In this section, we describe experiments to predict contact maps for objects based on their shape. ContactDB is the first large scale dataset that enables training data-intensive deep learning models for this task. Since ContactDB includes diverse contact maps for each object, the mapping from object shape to contact map is one-to-many and makes the task challenging. We explore two representations for object shape: single-view RGB-D, and full 3D. Since the contact patterns are significantly influenced by the functional intent, we train separate models for 'hand-off' and 'use'.

2.5.1 Single-view Prediction

Object shape is represented by an RGB-D image, and a 2D contact map is predicted for the visible part of the object. A single view might exclude information about important aspects of the object shape, and 'interesting' parts of the contact map might lie in the unseen half of the object. However, this representation has the advantage of being easily applicable to real-world robotics scenarios where mobile manipulators

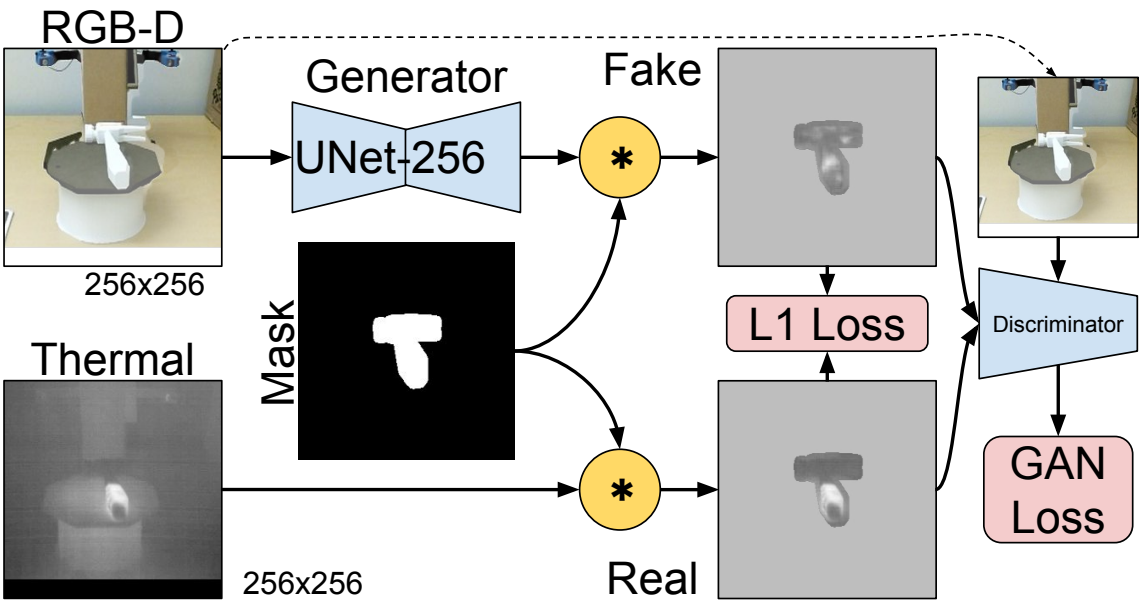


Figure 2.7: Training procedure for single-view contact map prediction. The discriminator has 5 conv layers followed by batch norm and leaky ReLU.

are often required to grasp objects after observing them from a single view. We used generative adversarial network (GAN)-based image-to-image translation [47, 48, 49] for this task, since the optimization procedure of conditional GANs is able to model a one-to-many input-output mapping [50, 51].

Figure 2.7 shows our training procedure and network architecture, which has roughly 54M and 3M parameters in the generator and discriminator respectively. We modified pix2pix [47] to accept a 4-channel RGB-D input and predict a single-channel contact map. The RGB-D stream from object scanning was registered to the thermal images, and used as input. Thermal images were used as a proxy for the single-view contact map. To focus the generator and discriminator on the object, we cropped a 256×320 patch around the object and masked all images by the object silhouette. All images from mug, pan, and wineglass were held out and used for testing. Figure 2.8 shows some predicted contact maps for these unseen objects, selected for looking realistic. Mug predictions for use have finger contact on the handle, whereas contact is observed over the top for handoff. Pan use predictions show grasps at the handle,

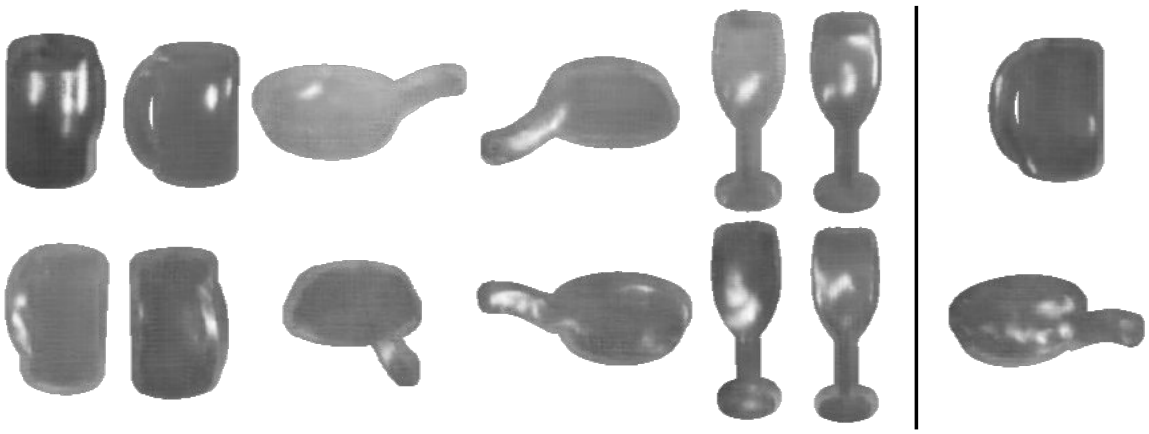


Figure 2.8: Single-view predictions from the pix2pix model for three *unseen* object classes: mug, pan and wine glass. Top: handoff intent, bottom: use intent. Rightmost column: uninterpretable predictions.

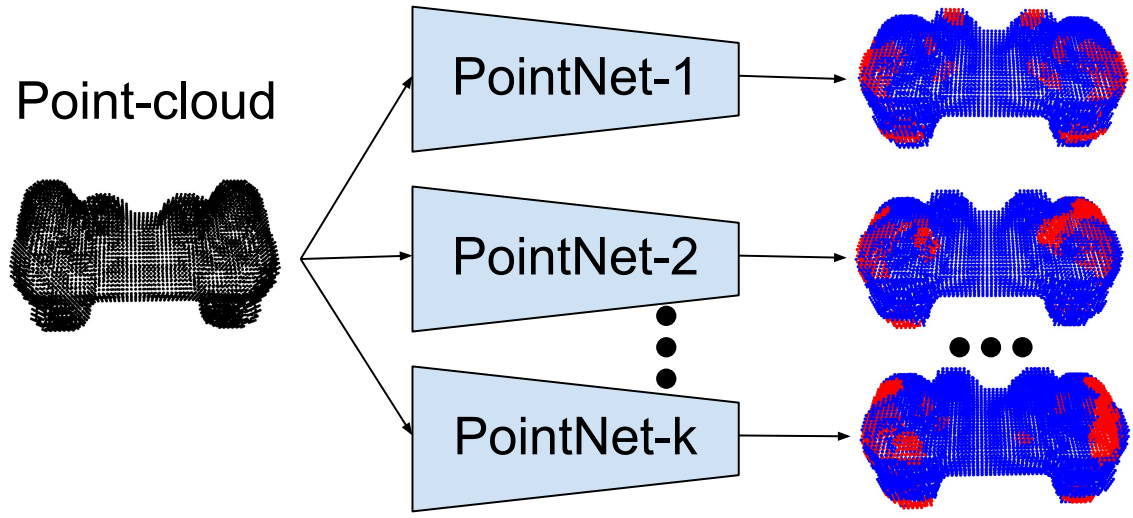
while handoff predictions additionally show a bi-manual grasp of the handle and side. Similarly, the wine glass indicates contact with a side grasp for use and over the opening for handoff.

2.5.2 3D Prediction

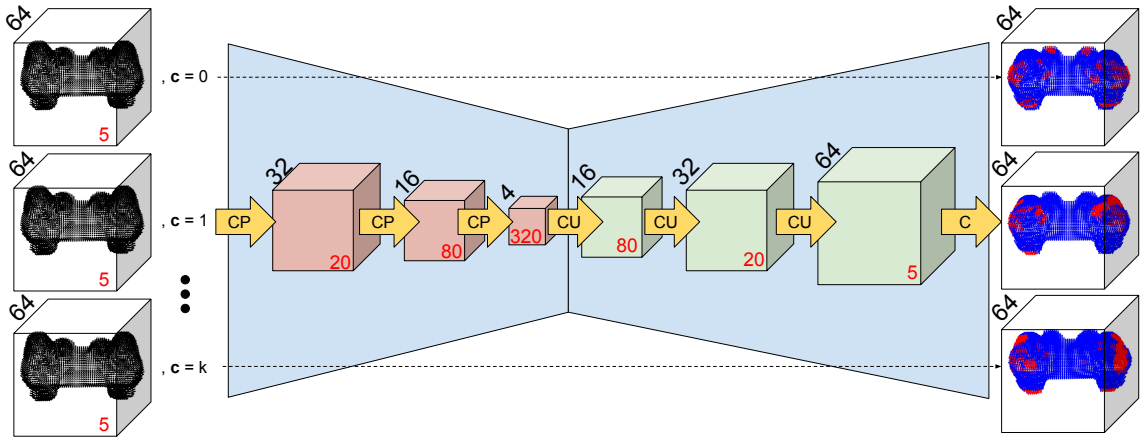
Full 3D representation gives access to the entire shape of the object, and alleviates the view-consistency problems observed during single-view prediction.

Learning a one-to-many-mapping. Stochastic Multiple Choice Learning [28] (sMCL) trains an ensemble of k predictors to generate k contact maps for each input (see Figure 2.9a). Each input has multiple equally correct ground truth maps. During training, the loss is backpropagated from each ground truth contact map to the network that makes the prediction closest to it. To encourage all members of the ensemble to be trained equally, as mentioned in [54], we made this association soft by routing the gradient to the closest network with a 0.95 weight and distributed the rest equally among other members of the ensemble, and randomly dropped entire predictions with a 0.1 probability. We trained models with $k = 1$ and $k = 10$.

In contrast, DiverseNet [29] generates diverse predictions from a single predictor



(a) sMCL with a PointNet predictor



(b) DiverseNet with a VoxNet predictor. CP: 3^3 conv with batch norm, ReLU and max pooling, CU: 3^3 conv with batch norm, ReLU and nearest neighbor upsampling. Black numbers: size of voxel grid, red numbers: number of channels.

Figure 2.9: 3D data representations and training strategies for predicting diverse contact maps. sMCL [28] requires multiple instances of a network, while DiverseNet [29] uses a single instance with an integer valued control variable. PointNet [52] operates on unordered point-clouds, whereas VoxNet [53] uses voxel occupancy grids.

Table 2.3: Diverse 3D contact map prediction errors (%) for the models presented in Section 2.5.2. Errors were calculated by matching each ground truth contact map with the closest from k diverse predictions, discarding predictions with no contact. '-' indicates that no contact was predicted.

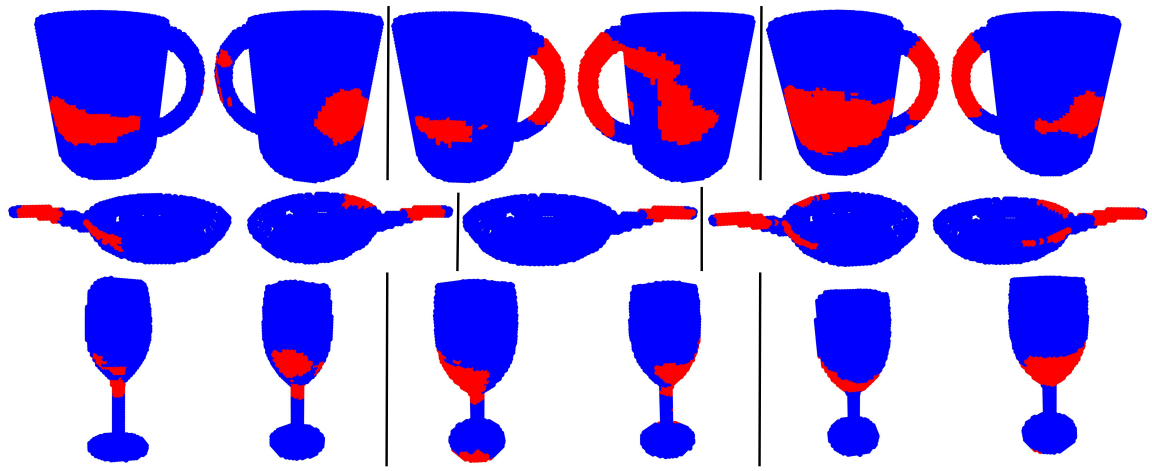
Test object	Handoff						Use					
	sMCL ($k=1$)		sMCL ($k=10$)		DiverseNet ($k=10$)		sMCL ($k=1$)		sMCL ($k=10$)		DiverseNet ($k=10$)	
	VoxNet	PointNet	VoxNet	PointNet	VoxNet	PointNet	VoxNet	PointNet	VoxNet	PointNet	VoxNet	PointNet
pan	76.80	-	**7.13**	20.43	8.48	19.68	17.22	-	8.25	43.57	**5.12**	22.58
wine glass	59.37	-	**11.11**	14.59	28.69	17.28	50.18	-	11.06	14.79	13.98	**10.47**
mug	29.93	-	16.68	27.10	**15.77**	21.60	66.03	-	32.51	31.30	**7.06**	32.41
average	55.37	-	**11.64**	20.71	17.65	19.52	44.48	-	17.27	29.89	**8.72**	21.82

network by changing the value of a one-hot encoded control variable **c** that is concatenated to internal feature maps of the network (See Figure 2.9b). Each ground truth contact map is associated with the closest prediction and gradients are routed through the appropriate **c** value. Diverse predictions can be generated at test time by varying **c**. Compared to sMCL, DiverseNet requires significantly fewer trainable parameters. We used 10 one-hot encoded **c** values in our experiments.

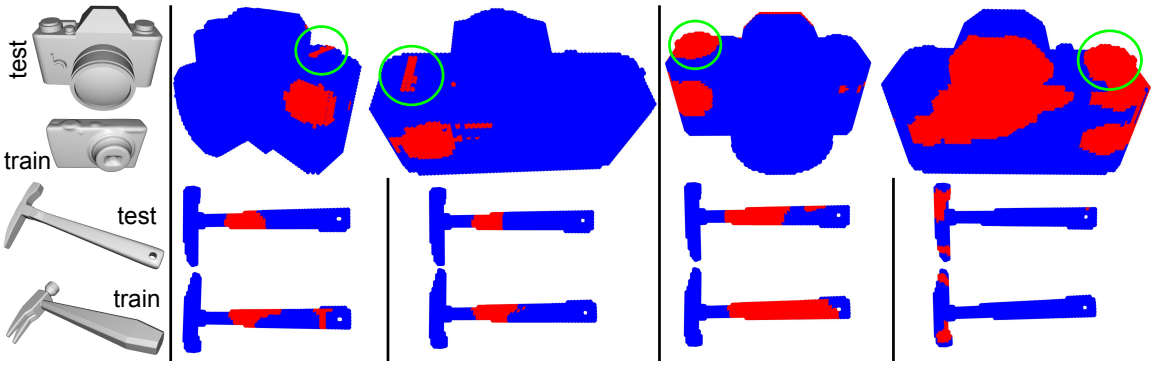
3D representation. We represented the 3D object shape in two forms: pointcloud and voxel occupancy grid. PointNet [52] operates on a pointcloud representation of the object shape, with points randomly sampled from the object surface. We normalized the XYZ position of each point to fit the object in a unit cube. The XYZ position and the normalization scale factor were used as 4-element features for each point. The network was trained by cross entropy loss to predict whether each voxel is in contact. We used a PointNet architecture with a single T-Net and 1.2M parameters.

VoxNet [53] operates on a solid occupancy grid of the object in a 64^3 voxelized space, and predicts whether each voxel is contacted. It uses 3D convolutions to learn shape features. The four features used for PointNet were used in addition to the binary occupancy value to form a 5-element feature vector for each voxel. Cross entropy loss was enforced only on the voxels on the object surface. The network architecture is shown in Figure 2.9b, and has approximately 1.2M parameters.

Experiments We conducted experiments with both VoxNet and PointNet, using



(a) Contact map predictions for unseen object classes



(b) Contact map predictions for an unseen shape of training object classes

Figure 2.10: Two views of diverse 3D contact map predictions. (a) *Unseen* object classes: mug, pan, and wine glass, (b) *Unseen shape* of training object classes: camera and hammer. Intent: use, Model: VoxNet-DiverseNet, Red: contact.

the sMCL and DiverseNet strategies for learning a one-to-many-mapping. For DiverseNet, we concatenated $\mathbf{c}$ to the output of the first and fifth conv layers in VoxNet, and to the input transformed by T-Net and the output of the second-last MLP in PointNet. Voxelization of the meshes was done using the algorithm of [55] implemented in binvox [56]. The PointNet input was generated by randomly sampling 3000 points from the object surface. We thresholded the contact maps at 0.4 after applying the sigmoid described in Section 2.4, to generate ground truth for classification. We augmented the dataset by randomly rotating the object around the yaw axis. PointNet input was also augmented by randomly choosing an axis and scaling the points along that axis by a random factor in [0.6, 1.4]. Dropout with $p = 0.2$ was applied to VoxNet-DiverseNet input. We found that similar dropout did not improve results for other models. Random sampling of surface points automatically acts like dropout for PointNet models, and sMCL models already incorporate a different dropout strategy as mentioned in Section 2.5.2. The cross entropy loss for contacted voxels was weighted by a factor of 10, to account for class imbalance. All models were trained with SGD with a learning rate of 0.1, momentum of 0.9 and weight decay of 5e-4. Batch size was 5 for models with $k = 10$, and 25 for models with $k = 1$.

Table 2.3 shows results on held-out test objects (mug, pan and wine glass). We conclude that the voxel occupancy grid representation is better for this task, and that a model limited to making a single prediction does not capture the complexity in ContactDB. Figures 2.10a and 2.10b show some of the 'use' intent predictions for unseen object classes and unseen shapes of training object classes respectively, selected for looking realistic. Mug predictions show horizontal grasps around the body. Predictions for the pan are concentrated at the handle, with one grasp being bimanual. Wine glass predictions show grasps at the body-stem intersection. Camera predictions show contact at the shutter button and sides, while predictions for the hammer show contact at the handle (and once at the head).

2.6 Conclusion and Future Work

We presented ContactDB, the first large-scale dataset of contact maps from functional grasping, analyzed the data to reveal interesting aspects of grasping behavior, and explored data representations and training strategies for predicting contact maps from object shape. We hope to spur future work in multiple areas. Contact patterns could inform the design of soft robotic manipulators by aiming to be able to cover object regions touched by humans. Research indicates that in some situations hand pose can be guided by contact points [44, 57]. Using contact maps to recover and/or assist in predicting the hand pose in functional grasping is an exciting problem for future research.

Acknowledgements: We thank Varun Agrawal for lending the 3D printer, Ari Kapusta for initial discussions on thermal cameras, NVIDIA for a GPU grant, and all the anonymous participants involved in data collection.

CHAPTER 3

CONTACTPOSE: A DATASET OF GRASPS WITH OBJECT CONTACT AND HAND POSE

Abstract: Grasping is natural for humans. However, it involves complex hand configurations and soft tissue deformation that can result in complicated regions of contact between the hand and the object. Understanding and modeling this contact can potentially improve hand models, AR/VR experiences, and robotic grasping. Yet, we currently lack datasets of hand-object contact paired with other data modalities, which is crucial for developing and evaluating contact modeling techniques. We introduce ContactPose, the first dataset of hand-object contact paired with hand pose, object pose, and RGB-D images. ContactPose has 2265 unique grasps of 25 household objects grasped with functional intents by 50 participants. Analysis of ContactPose data reveals interesting relationships between hand pose and contact. We use this data to rigorously evaluate various data representations, heuristics from the literature, and learning methods for contact modeling.

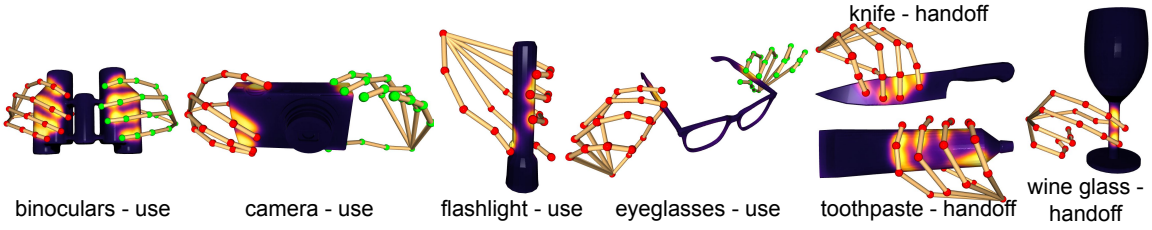


Figure 3.1: Examples from ContactPose, a dataset capturing grasps of household objects. ContactPose includes high-resolution contact maps (object meshes texture-mapped with hand-object contact), 3D joint locations, and multi-view RGB-D videos of grasps (not shown here). Left hand joints are **green**, right hand joints are **red**.

3.1 Introduction

A person's daily experience includes numerous and varied hand-object interactions. Understanding and reconstructing hand-object interaction has received growing attention from the computer vision, computer graphics, and robotics communities. Most research has focused on hand pose estimation [58, 59, 60, 61], realistic hand and body reconstruction [62, 63, 64, 65], and robotic grasp prediction for anthropomorphic hands [66, 67]. Here, we address the under-explored problem of *hand-object contact modeling*, including predicting which points on the object are in contact with the hand based on other information about the grasp, such as the 3D pose of the hand and images of the grasp. Accurate contact models have numerous applications in computer interfaces, understanding social interaction, object manipulation, and safety. For example, a hand contact model could interpret computer commands from physical interactions with a 3D printed replica object, or estimate if pathogens from a contaminated surface were transmitted through contact. More broadly, accurate contact modeling can improve estimation of grasp dynamics [68, 69, 70, 71], which can lead to better VR simulations of grasping scenarios and grasping with soft robotic hands [72, 73].

Lack of ground-truth data has likely played a role in the under-exploration of this problem. Typically, the contacting surfaces of a grasp are occluded from direct observation with visible light imaging. Approaches that instrument the arm or hand with sensorized gloves [74, 75] can subtly influence natural grasping behavior, and do not measure contact on the object surface. Approaches that intersect hand models with object models to infer contact require careful selection of proximity thresholds or specific contact points on the hand [63, 62]. In addition, they cannot account for the effects of soft hand tissue deformation, since existing state-of-the-art hand models [76] are rigid.

Brahmbhatt *et al.* [77] recently introduced thermal cameras as sensors for capturing detailed ground-truth contact. Their method observes the heat transferred from the (warm) hand to the object through a thermal camera after the grasp. We adopt their method in ContactPose because it avoids the pitfalls mentioned above and allows for evaluation of contact modelling approaches with ground-truth data. However, it also imposes some constraints. 1) The objects have a plain visual texture since they are 3D printed to ensure consistent thermal properties. This does not affect contact modeling methods that rely on 3D shape and not texture, like 3D hand pose-based methods and many practical applications like simulations for VR and robotic grasping. It does limit the generalization ability of RGB image-based methods, which can potentially be mitigated by use of depth images and synthetic textures. 2) The grasps are static, because in-hand manipulation results in multiple overlapping thermal hand-prints that depend on timing and other factors. Contact modeling for static grasps is still an unsolved problem, and forms the basis for future work on dynamic grasps. The contact modeling methods we present here could be applied to dynamic scenarios on a frame-by-frame basis.

In addition, we develop a data collection protocol that captures multi-view RGB-D videos of the grasp, and an algorithm for 3D reconstruction of hand joints (§ 3.3.1). To summarize, we make the following contributions:

- **Data**: We introduce ContactPose, a dataset that captures 50 participants each grasping 25 objects with two different functional intents. In addition to high-quality contact maps for each grasp, it includes over 1.5 M RGB-D images from 3 viewpoints, with object pose and 3D hand joints annotated in each frame. We will make this datset available for public use to encourage research in contact modelling, and in contact-aware hand- and object-pose estimation broadly.

- **Analysis**: We dissect this data in various ways to explore the interesting relationship between contact and hand pose. This reveals some surprising patterns,

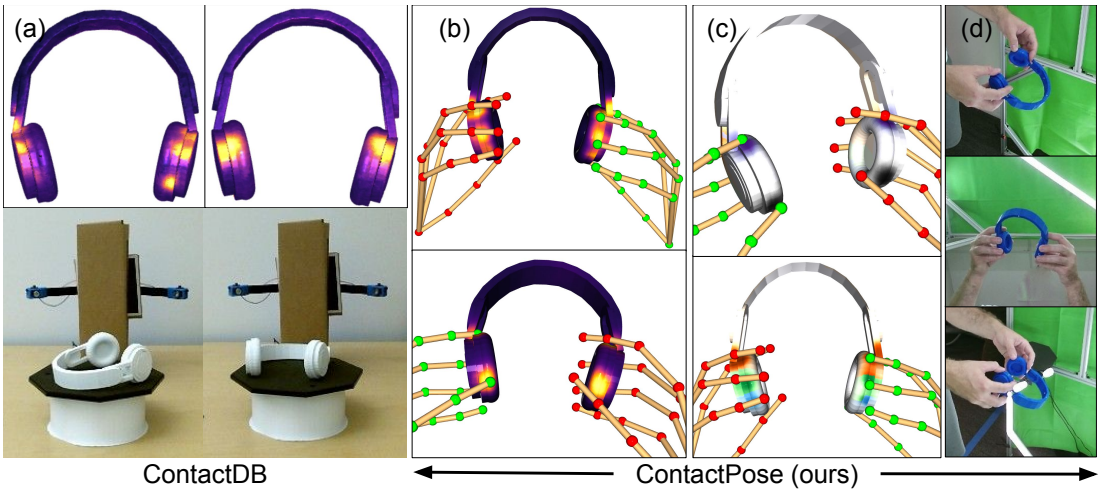


Figure 3.2: Comparison to ContactDB [77]. It includes contact maps and turntable RGB-D images (a), which are often not enough to fully interpret the grasp e.g. it is not clear which fingers generated the contact. In contrast, ContactPose includes 3D joint locations (b), which allows association of contacted areas to hand parts (c), and multi-view RGB-D grasp images (d). These data enable a more comprehensive interpretation of the grasp.

and confirms some common intuitions.

- **Algorithms**: We explore various representations of object shape, hand pose, contact, and network architectures for learning-based contact modeling. Importantly, we rigorously evaluate these methods (and heuristic methods from the literature) against ground-truth unique to ContactPose.

3.2 Related Work

Capturing and modeling contact: Previous works have instrumented hands and/or objects to capture contact. Sundaram *et al.* [75] used a tactile glove to capture hand contact during grasping. Brahmbhatt *et al.*[77] used a thermal camera after the grasp to observe the heat residue left by the warm hand on the object surface. However, these methods did not capture hand pose or grasp images, which are necessary for developing applicable contact models (Figure 3.2). Pham *et al.*[80] tracked hands and simple objects in videos, and trained models to predict contact forces at fingertips

Table 3.1: Comparison with existing hand-object interaction datasets. ContactPose (ours) stands out for its size, and paired hand-object contact, hand pose and object pose.

Feature	**FPHA** [60]	**HO-3D** [78]	**FreiHand** [79]	**STAG** [75]	**ContactDB** [77]	**Ours**
3D joints	✓	✓	✓	×	×	✓
Object pose	✓	✓	×	×	✓	✓
Grasp RGB images	✓	✓	✓	✓	×	✓
Grasp Depth images	✓	✓	×	×	×	✓
Natural hand appearance	×	✓	✓	×	×	✓
Natural object appearance	×	✓	✓	✓	×	×
Naturally situated	✓	×	×	×	×	×
Multi-view images	×	×	✓	×	×	✓
Functional intent	✓	×	×	×	✓	✓
Hand-object contact	×	×	×	✓	✓	✓
# Participants	6	8	32	1	50	50
# Objects	4	8	35	26	50	25

that explain the motion, which are evaluated against embedded force transducer data from sparse object points, in [81]. In contrast, we focus on detailed contact modeling for complex objects and grasps, evaluated against contact maps over the entire object surface.

Contact heuristics: Some heuristic methods have been proposed to detect hand-object contact, often aimed at improving hand pose estimation. Hamer *et al.*[24] performed joint hand tracking and object reconstruction, and inferred contact only at fingertips using proximity threshold. In simulation [82] and robotic grasping [70, 83], contact is often determined similarly, or through collision detection [84, 85]. Ballan *et al.*[86] defined a cone circumscribing object mesh triangles, and penalized penetrating hand points (and vice versa). This formulation has also been used to penalize self-penetration and environment collision [62, 87]. While such methods were evaluated only through proxy tasks (*e.g.* hand pose estimation), ContactPose enables evaluation against ground-truth contact (§ 3.6).

Grasp Datasets: Focusing on datasets involving hand-object interaction, hand pose has been captured in 3D with magnetic trackers [60], optimization [78], multi-view boot-strapping [59], semi-automated human-in-the-loop [79], manually [57], synthetically [63], or as instances of a taxonomy [88, 89, 21] along with RGB-D images

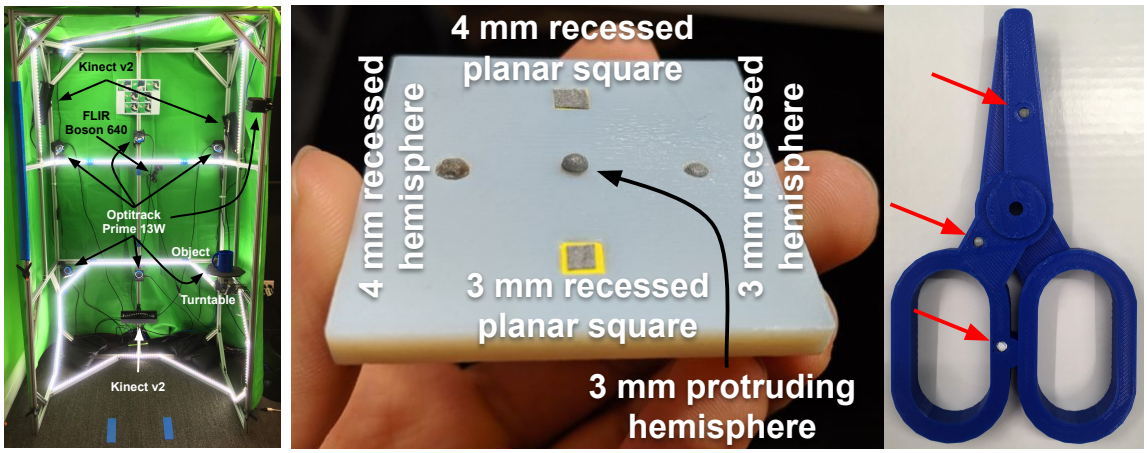
depicting the grasps. However, none have contact annotations (see Table 3.1), and suffer additional drawbacks like lack of object information [79, 59] and simplistic objects [57, 60] and interactions [57, 63], which make them unsuitable for our task. In contrast, ContactPose has ground-truth contact, and real RGB-D images of complex (including bi-manual) functional grasps for complex objects. The plain object texture is a drawback of ContactPose. Tradeoffs for this in the context of contact modelling are discussed in § 3.1.

3.3 The ContactPose Dataset

In ContactPose, hand-object contact is represented as a contact map on the object mesh surface, and observed through a thermal camera. Hand pose is represented as 3D hand(s) joint locations in the object frame, and observed through multi-view RGB-D video clips. The cameras are calibrated and object pose is known, so that the 3D joints can be projected into images (examples shown in Appendix B). Importantly, we avoid instrumenting the hands with data gloves, magnetic trackers or other sensors. This has the dual advantage of not interfering with natural grasping behavior and allowing us to use the thermal camera-based contact capture method from [77]. We develop a computational approach (Section 3.3.2) that optimizes for the 3D joint locations by leveraging accurate object tracking and aggregating over multi-view and temporal information. Our data collection protocol, described below, facilitates this approach.

3.3.1 Data Capture Protocol and Equipment

We invite able-bodied participants to our laboratory and collect data through the following IRB-approved protocol. Objects are placed at random locations on a table in orientation normally encountered in practice. Participants are instructed to grasp an object with one of two functional intents (either using the object, or handing it



(a) (b)

Figure 3.3: (a) Our setup consists of 7 Optitrack Prime 13W tracking cameras, 3 Kinect v2 RGB-D cameras, a FLIR Boson 640 thermal camera, 3D printed objects, and a turntable. (b) **Left**: Different object tracking marker configurations we investigate. **Right**: 3D printed object with recessed 3 mm hemispherical markers (highlighted by **red** arrows) offer a good compromise between unobtrusiveness and tracking performance.

off). Next, they stand in the data collection area (Figure 3.3a) and move the object for 10-15 s in the cubical space. They are instructed to hold their hand joints steady, but are free to arbitrarily rotate the wrist and elbow, and to grasp objects with both hands or their dominant hand. This motion is recorded by 3 Kinect v2 RGB-D cameras (used for hand pose) and an Optitrack motion capture (mocap) system (used for object pose). Next, they hand the object to a researcher, who places it on a turntable, handling it with gloved hands. The object is recorded with the mocap system, Kinect v2, and a FLIR Boson 640 thermal camera as the turntable rotates a circle. Thermal images are texture-mapped to the object mesh to construct the contact maps.

Object Selection and Fabrication: We capture grasps on a subset of 25 objects from [77] that are applicable for both 'use' and 'hand-off' functional grasping (see Appendix B for a list). The objects are 3D printed in blue for good contrast with hands and the green background of our capture area. 3D printing the objects ensures

consistent thermal properties and ensures geometric consistency between real world objects in capture sessions and the 3D models in our dataset.

Mocap recovers the object pose using retro-reflective markers, whose the placement on the object requires some care. Attaching a large 'marker tree' would block interactions with a significant area of the surface. Placing hemispherical markers on the surface is more promising, but a sufficient number (8+) are needed to ensure visibility during hand occlusion and the resulting 'bumps' can be uncomfortable to touch, which might influence natural grasping behavior. We investigate a few alternative marker configurations (Figure 3.3b). Flat pieces of tape were more comfortable but only tracked well when the marker was directly facing the camera. A good compromise is to use 3 mm hemispherical markers but to recess them into the surface by adding small cut-outs during 3D printing. These are visible from a wide range of angles but do not significantly affect the user's grip. Fixing the marker locations also allows for simple calibration between the Optitrack rigid body and the object's frame.

3.3.2 Grasp Capture without Hand Markers

Each grasp is observed through N time frames, each containing RGB-D images from C cameras. We want to estimate the 3D joint locations in every frame. Assuming that the hand pose relative to the object is fixed, and given the 6-DOF object pose for each frame, we aggregate the noisy per-frame 2D joint detections into a single set of high-quality 3D joint locations, which can be transformed by the frame's object pose.

For each RGB frame, we use Detectron [90] to locate the wrist, and run the OpenPose hand keypoint detector [59] on a 200×200 crop around the wrist. This produces 2D joint detections $\{\mathbf{x}^{(i)}\}_{i=1}^{N}$ and confidence values $\{\mathbf{w}^{(i)}\}_{i=1}^{N}$, following the 21-joint format from [59]. One option is to lift these 2D joint locations to 3D using

the depth image [58], but that biases the location toward the camera and the hand surface (our goal is to estimate joint locations internal to the hand). Furthermore, the joint detections at any given frame are unreliable. Instead, we use our hand-object rigidity assumption to estimate the 3D joint locations ${}^{\mathbf{o}}\mathbf{X}$ in the object frame that are consistent with all NC images. This is done by minimizing the average re-projection error:

$$\min_{{}^{\mathbf{o}}\mathbf{X}} \sum_{i=1}^{N} \sum_{c=1}^{C} \mathcal{D}\left(\mathbf{x}_c^{(i)}, \pi\left({}^{\mathbf{o}}\mathbf{X}; K_c, {}^{c}T_w \; {}^{w}T_o^{(i)}\right); \mathbf{w}_c^{(i)}\right) \tag{3.1}$$

where $\mathcal{D}$ is a distance function, and $\pi(\cdot)$ is the camera projection function using camera intrinsics K_c and object pose w.r.t. camera at frame i, ${}^{c}T_o^{(i)} = {}^{c}T_w \; {}^{w}T_o^{(i)}$. Our approach requires the object pose w.r.t. world at each frame ${}^{w}T_o^{(i)}$ i.e. object tracking. This is done using an Optitrack motion capture system tracking markers embedded in the object surface.

In practice, the 2D joint detections are noisy and object tracking fails in some frames. We mitigate this by using the robust Huber function [91] over Mahalanobis distance ($\mathbf{w}^{(i)}$ acting as variance) as $\mathcal{D}$, and wrapping Eq. 3.1 in a RANSAC [92] loop. A second pass targets frames that fail the RANSAC inlier test due to inaccurate object pose. Their object pose is estimated through the correspondence between their 2D detections and the RANSAC-fit 3D joint locations, and they are included in the inlier set if they pass the inlier test (re-projection error less than a threshold). It is straightforward to extend the optimization described above to bi-manual grasps. We get a low re-projection error of 3-5 pixels w.r.t. (inherently noisy) 2D joint detections over > 100 3-view frames for each grasp, indicating that participants indeed followed the static grasp instruction. We manually curated the dataset, including clicking 2D joint locations to aid the 3D reconstruction in some cases, and discarding some obviously noisy data.

Hand Mesh Models: In addition to capturing grasps, hand shape information is collected through palm contact maps on a flat plate, and multi-view RGB-D videos

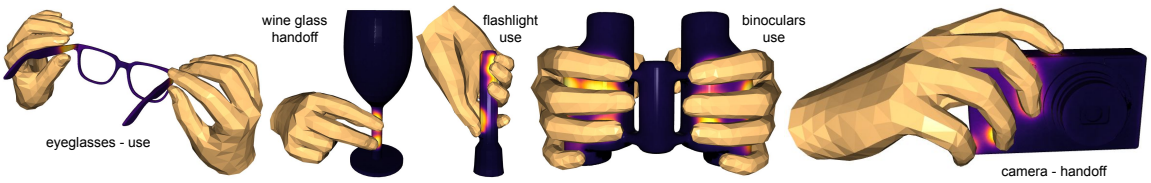


Figure 3.4: MANO hand meshes [76] fit to ContactPose data. Both hand pose and shape parameters are optimized to minimize the distance of MANO joints from ContactPose 3D joint annotations.

of the participant performing 7 known hand gestures (shown in Appendix B). Along with accurate 3D joints, this data enables fitting of the MANO hand mesh model [76] to each grasp (Figure 3.4). We use these hand meshes for some of the analysis and learning experiments discussed below.

3.4 Data Analysis

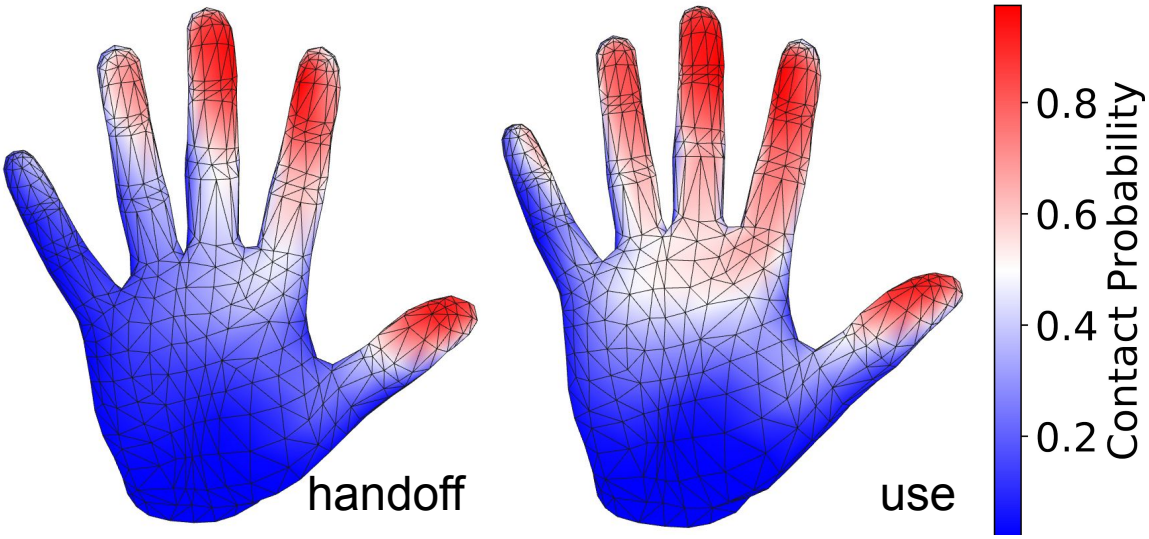
All contact maps are normalized to $[0, 1]$ following the sigmoid fitting procedure from [77].

3.4.1 Association of Contact to Hand Parts

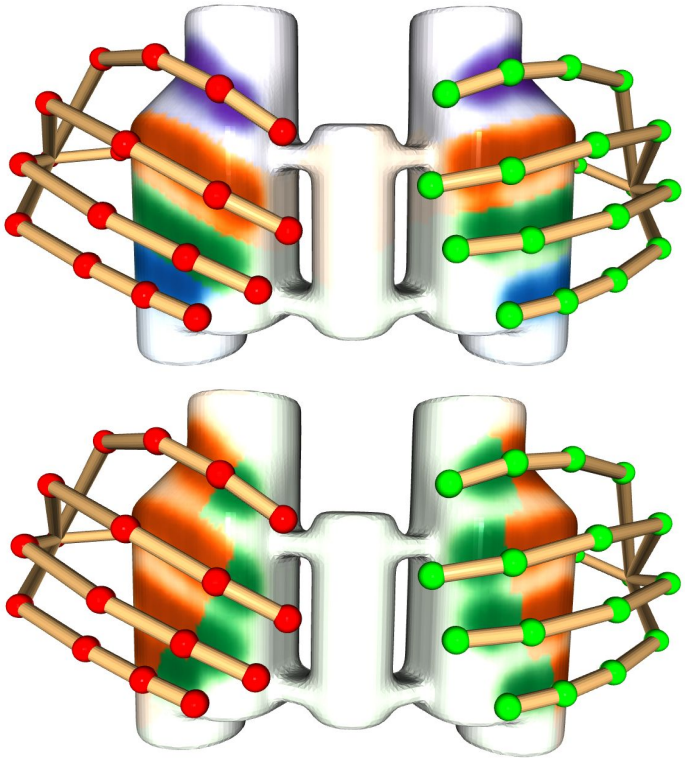
It has been observed that certain fingers and parts of fingers (e.g. fingertips) are more frequently in contact with the object than other parts [89, 93]. ContactPose allows us to quantify this. This can potentially inform grasp synthesis, anthropomorphic robotic hand design, and tactile sensor (e.g. BioTac [94]) placement in robotic hands.

For each grasp, we threshold the contact map at 0.4 and associate each contacted object point with its nearest hand point from the fitted MANO hand mesh. A hand point is considered to be in contact if one or more contacted object points are associated with it. A coarser analysis at the phalange level is possible by modeling phalanges as line segments connecting joints. In this case, the distance from an object point to a phalange is the distance to the closest point on the line segment.

Figure 3.5a shows the contact probabilities averaged over 'use' and 'hand-off'



(a)



(b)

Figure 3.5: (a) Hand contact probabilities estimated from the entire dataset. (b) Association of contacted binoculars points with fingers (top) and sets of phalanges at the same level of wrist proximity (bottom), **indicated by different colors**.

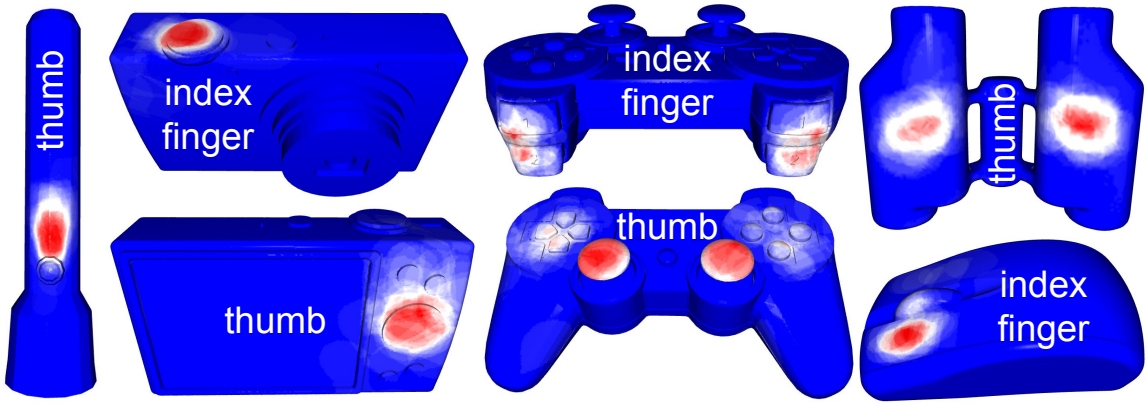


Figure 3.6: Automatic 'active area' discovery: Contact probability for various hand parts on the object surface.

grasps. Not surprisingly, the thumb, index, and middle finger are the most contacted fingers, and tips are the most contacted phalanges. Even though fingertips receive much attention in grasping literature, the contact probability for all three phalanges of the index finger is *higher* than the contact probability of the pinky fingertip. Proximal phalanges and palm also have significant contact probabilities. This is consistent with observations made by Brahmbhatt et al [77]. Interestingly, contact is more concentrated at the thumb and index finger for 'hand-off' than 'use'. 'Use' grasps have an average contact area of 35.87 cm^2 compared to 30.58 cm^2 for 'hand-off'. This analysis is similar to that in Fig. 3 of Hasson *et al.*[63], but supported by ground-truth contact rather than synthetic grasps.

3.4.2 Automatic Active Area Discovery

Brahmbhatt et al [77] define active areas as regions on the object highly likely to be contacted. While their analysis was limited to manually selecting active areas and measuring their probability of being contacted by any part of the hand, ContactPose allows us to 'discover' active areas automatically and for specific hand parts. We use the object point-phalange association from § 3.4.1 (*e.g.* Fig. 3.5b) to estimate the probability of each object point being contacted by a given hand part (e.g. index

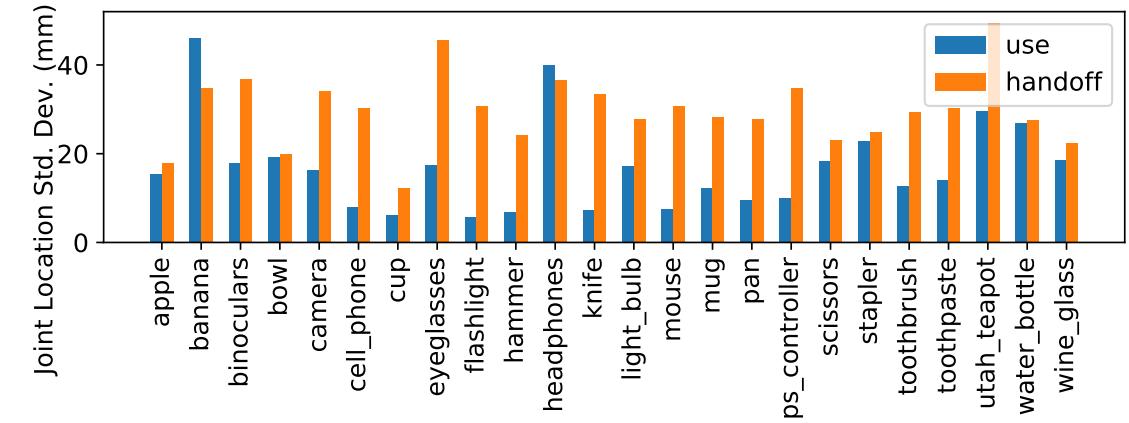
finger tip), which can be thresholded to segment the active areas. Figure 3.6 shows this probability for the index fingertip and thumb, for 'use' grasps of some objects. This could potentially inform locations for placing contact sensors (real [81] or virtual for VR) on objects.

3.4.3 Grasp Diversity

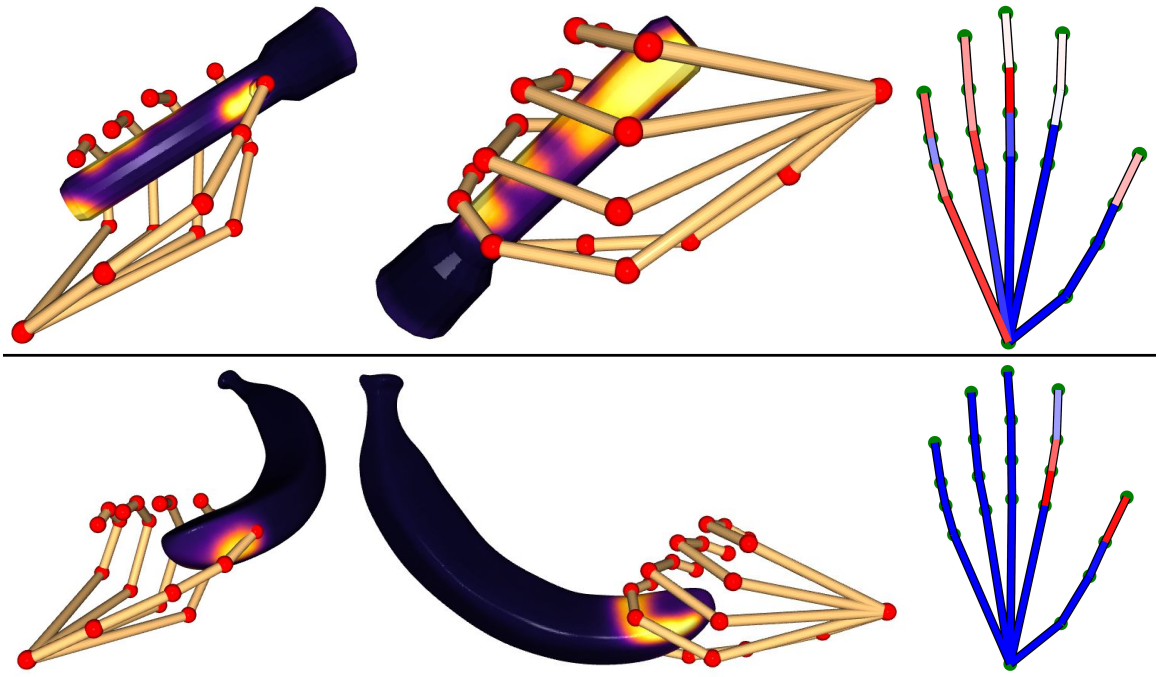
We further quantify the effect of intent on grasping behavior by measuring the standard deviation of 3D joint locations over the dataset. The mean of all 21 joint standard deviations is shown in Figure 3.7a. It shows that 'hand-off' grasps are more diverse than 'use' grasps in terms of hand pose. We accounted for symmetrical objects (e.g. wine glass) by aligning the 6 palm joints (wrist + 5 knuckles) of all hand poses for that object to a single set of palm joints, where the only degree of freedom for alignment is rotation around the symmetry axis. Hand size is normalized by scaling all joint location such that the distance from wrist to middle knuckle is constant.

Organizing the grasps by clustering these aligned 3D joints (using L2 distance and HDBSCAN [95]) reveals the diversity of grasps captured in ContactPose (Figure 3.8). 'Hand-off' grasps exhibit a more continuous variation than 'use' grasps, which are tied more closely to the function of the object.

Figure 3.7b shows pair of grasps found by minimizing hand pose distance and maximizing hand contact distance over the entire dataset. We use the phalange-level contact association described in § 3.4.1. Summing the areas of all object mesh triangles incident to all vertices associated with a phalange creates a 20-dimensional vector. We use L2 distance over this vector as contact distance. Figure 3.7b shows that grasps with similar hand pose can contact different parts of the object and/or hand, inducing different forces and manipulation possibilities [60] and emphasizing that hand pose alone provides an inadequate representation of grasping.



(a)



(b)

Figure 3.7: (a) Per-object standard deviation in 3D joint locations, for 'use' and 'hand-off'. 'Hand-off' grasps consistently exhibit more diversity than 'use' grasps. (b) A pair of grasps with similar hand pose but different contact characteristics. Hand contact feature color-coding is similar to Figure 3.5a.

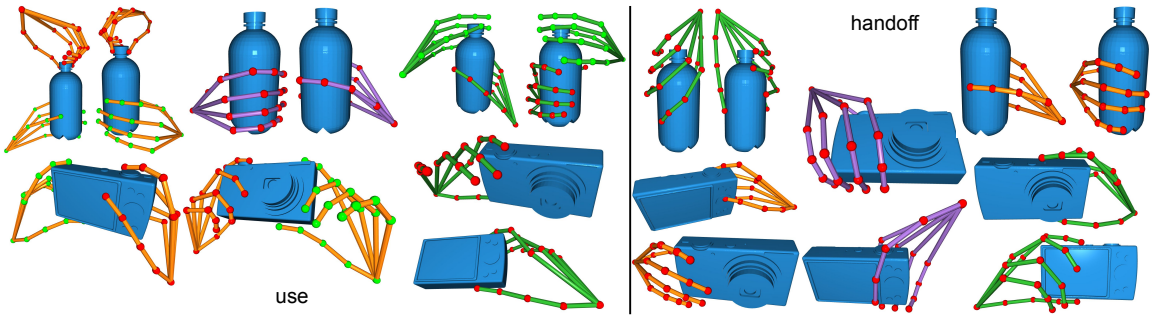


Figure 3.8: Examples from hand pose clusters for 'use' and 'hand-off' grasps. Grasps from different clusters are shown **with different colors** (some grasps are bi-manual). Left hand joints are **green**, right hand joints are **red**.

3.5 Contact Modeling Experiments

This section describes our experiments on *contact modeling* given the hand pose (3D joint locations) or RGB grasp image(s), assuming known object geometry and pose. Our experiments focus on finding good data representations and learning algorithms, and evaluating techniques against ground-truth.

Object Shape Representation: We represent the object geometry through either a pointcloud densely sampled from the object body (with 1K to 30K points based on the object size), or a 64^3 voxel occupancy grid. Features encoding the input hand pose are associated with individual points or voxels. The entire pointcloud or voxel grid is then processed to predict contact values for points or surface voxels.

Hand Pose Representation: Features relating object shape to hand pose are computed for each point or voxel. These features have varying levels of richness of hand shape encoding. To simulate occlusion and noisy pose perception for the first 4 features, we sample a random camera pose and drop (set to 0) all features associated with the farthest 15% of the joints from the camera.

- **`simple-joints`**: We start by simply using the 21 3D joint locations w.r.t. the object coordinate system as 63-dimensional features for every point. For bi-manual grasps, the hand with the closest joint to a point is used to provide

features for that point.

- **`relative-joints`**: Since contact at an object surface point depends on the *relative* position of the finger, we next calculate relative vectors from an object point to every joint of the hand closest to it. Contact also depends on the surface geometry: a finger is more likely to contact an object point if the vector to it is parallel to the surface normal at that point. Hence we use unit-norm surface normals and the relative joint vectors to form $63 + 3 = 66$-dimensional features for every point.

- **`skeleton`**: To better capture hand joint connectivity, we compute relative vectors from an object point to the nearest point on phalanges, modeled as line segments. 40-dimensional features for each object point are constructed by concatenating the lengths of 20 such vectors (one for each phalange), and their dot product with the surface normal at that object point.

- **`mesh`**: These features leverage the rich hand geometry present in the MANO hand model. A relative vector is constructed from the object point to its closest hand mesh point. 23-dimensional features are constructed from the length of this vector, its dot product with the surface normal, and distances to 21 hand joints.

- Grasp Image(s): To investigate if CNNs can extract relevant information directly from images, we extract dense 40-dimensional features from 256×256 crops of RGB grasp images using a CNN encoder-decoder inspired by U-Net [96] (see Appendix B for architecture). These images come from the same time instant. We investigate both 3-view and 1-view settings, with feature extractor being shared across views for the former. Features are transferred to corresponding 3D object points using the known object pose and camera intrinsics, averaging the features if multiple images observe the same 3D point (Figure 3.10a).

Points not visible from any image have all features set to 0. Image backgrounds are segmented by depth thresholding at the 20th percentile, and the foreground pixels are composited onto a random COCO [97] image. This investigation is complementary to recent work on image-based estimation of object geometry [98, 99], object pose [100, 101], and hand pose [59, 65, 61, 79, 78].

Contact Representation: We observed in early experiments that contact maps supervised with a mean squared error loss were blurred and saturated. We conjecture that this is due contact value occurrence imbalance and discontinuous contact boundaries for smooth input features. Hence, we discretize the $[0, 1]$ normalized values into 10 equal bins and treat contact prediction as a classification problem, inspired by the image colorization approach from Zhang et al [102]. We use the weighted cross entropy loss, where the weight for each bin is proportional to a linear combination of the inverse occurrence frequency of that bin and a uniform distribution (Eq. 4 from [102] with $\lambda = 0.4$). Following [102], we derive a point estimate for contact in $[0, 1]$ from classification outputs using the annealed mean ($T = 0.1$).

Learning Algorithms: Given the hand pose features associated with points or voxels, the entire pointcloud or voxel grid is processed by a neural network to predict the contact map. We use the PointNet++ [103] architecture implemented in pytorch-geometric [104, 105] (modified to reduce the number of learnable parameters) for pointclouds, and the VoxNet [106]-inspired 3D CNN architecture from [77] for voxel grids (see Appendix B for architectures). For voxel grids, a binary feature indicating voxel occupancy is appended to hand pose features. Since hand pose features are related to surface quantities, they are set to 0 for voxels inside the object. Because the features are rich and provide fairly direct evidence of contact, we include a simple learner baseline of a multi-layer perceptron (MLP) with 90 hidden nodes, parametric ReLU [107] and batchnorm [108].

Contact Modeling Heuristics: We also investigate the effectiveness of heuristic

Table 3.2: Contact prediction re-balanced AuC (%) (higher is better) for various combinations of features and learning methods.

Learner	Features	mug	pan	wine-glass	Average	Rank
None	Heuristic [86, 62]	78.47	83.06	81.79	81.11	3
VoxNet [106, 77]	skeleton	73.97	82.12	76.30	77.46	
MLP	simple-joints	74.69	79.89	73.68	76.09	
	relative-joints	73.20	79.70	75.91	76.27	
	skeleton	76.75	80.72	80.81	79.43	5
	mesh	81.29	**85.83**	82.52	**83.21**	1
PointNet++	simple-joints	74.64	73.32	67.92	71.96	
	relative-joints	73.89	74.91	74.88	74.56	
	skeleton	78.84	79.06	82.24	80.05	4
	mesh	**82.92**	83.13	**83.33**	83.13	2
Image enc-dec, PointNet++	images (3-view)	76.28	81.56	80.14	79.33	
	images (1-view)	71.28	78.71	72.64	74.21	

techniques, given detailed hand geometry through the MANO hand mesh. Specifically, we use the conic distance field Ψ from [86, 62] as a proxy for contact intensity. To account for imperfections in hand modelling (due to rigidity of the MANO mesh) and fitting, we compute Ψ not only for collisions, but also when the hand and object meshes are closer than 1 cm. Finally, we calibrate Ψ to our ground truth contact through least-squares linear regression on 4700 randomly sampled contact points. Both these steps improve the technique's performance.

3.6 Results

In this section, we evaluate various combinations of features and learning algorithms described in § 3.5. The metric for quantitative evaluation is the area under the curve formed by calculating accuracy at increasing contact difference thresholds. Following [102], this value is re-balanced to account for varying occurrence frequencies of values in the 10 contact bins. Following [77], we hold out all grasps for 3 objects (mug, pan and wine glass) for evaluation, and train our models on the rest.

Table 3.2 shows the re-balanced AuC values averaged over joint drop probability $\in [0, 0.3]$ and 3 runs for all the non-image-based variants. We observe that features

capturing richer hand shape information perform better (*e.g.* **simple-joints** vs. **skeleton** and **mesh**). Learning-based techniques with **mesh** features that operate on pointclouds are able to outperform heuristics, even though the latter has access to the full high-resolution object mesh, while the former makes predictions on a pointcloud. Learning also enables **skeleton** features, which have access to only the 3D joint locations, to perform competitively against mesh-based heuristics and features. While image-based techniques are not yet as accurate as the hand pose-based ones, a significant boost is achieved with multi-view inputs.

Figure 3.9 shows qualitative results for contact prediction from hand pose (predictions are transferred from the pointcloud to high-resolution meshes for better visualization). The **skeleton**-PointNet++ combination is able to predict plausible contact patterns for dropped-out parts of the hand, and capture some of the nuances of palm contact. The **mesh**-PointNet++ combination captures more nuances, especially at the thumb and bottom of the palm. In contrast, **relative-joints** features-based predictions are diffused, lack finer details, and have high contact probability in the gaps between fingers, possibly due to lack of access to information about joint connectivity and hand shape.

Figure 3.10b shows qualitative results for contact prediction from RGB images. These predictions have less high-frequency details compared to hand pose based predictions. They also suffer from depth ambiguity – the proximal part of the index appears to be in contact from the mug images, but is actually not. This can potentially be mitigated by use of depth images.

3.7 Conclusion and Future Work

We introduced ContactPose, the first dataset of paired hand-object contact, hand pose, object pose, and RGB-D images for functional grasping. Data analysis revealed some surprising patterns, like higher concentration of hand contact at the first three

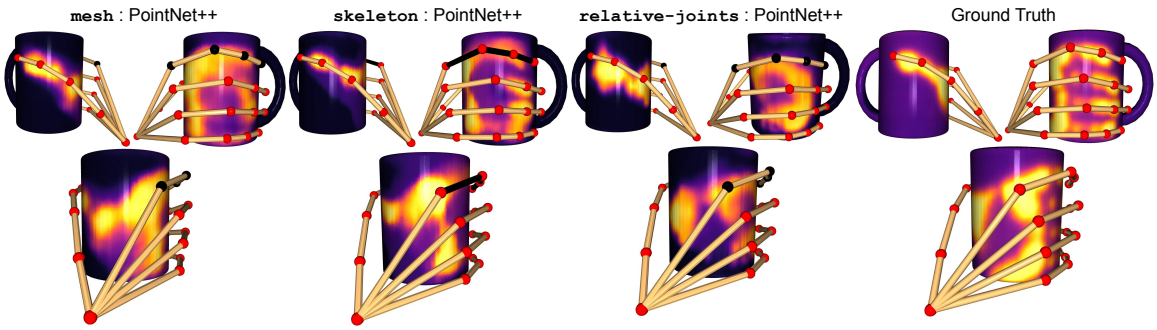
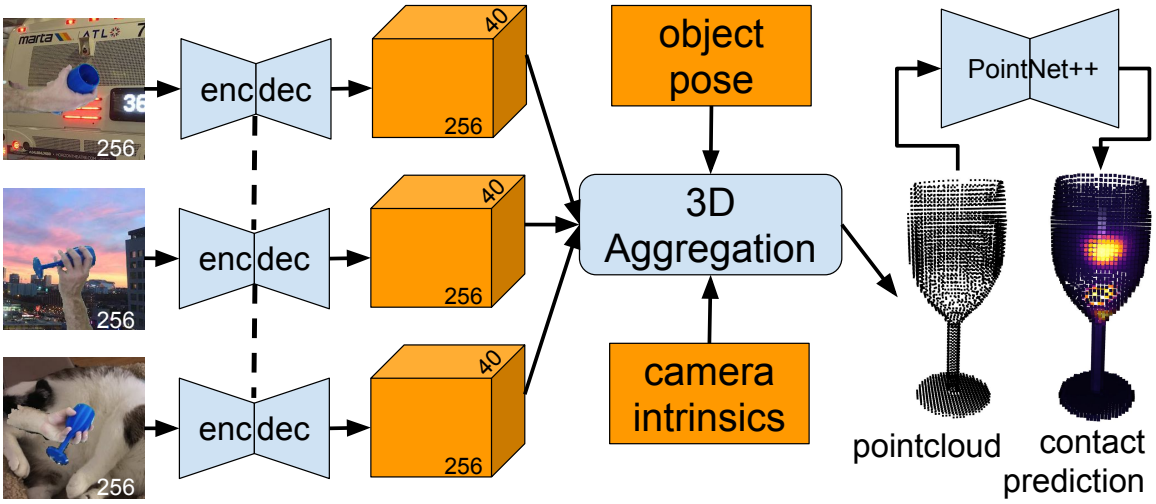


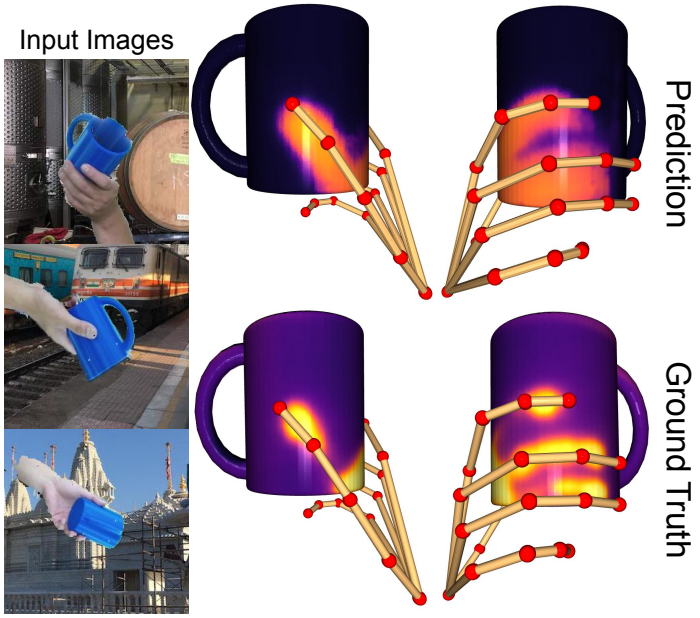
Figure 3.9: Contact prediction from hand pose. All input features related to black line segments and joints were dropped (set to 0). Notice how the **mesh**- and **skeleton**-PointNet++ predictors is able to capture nuances of palm contact, thumb and finger shapes.

fingers for 'hand-off' vs. 'use' grasps. We also showed how learning-based techniques for geometry-based contact modeling can capture nuanced details missed by heuristic methods.

An interesting direction for future work could be using this contact ground-truth to develop more realistic, deformable hand mesh models. State-of-the-art models (*e.g.* [76, 109]) are rigid, while the human hand is covered with soft tissue. As the Future Work section of [76] notes, they are trained with meshes from which objects are manually removed, and do not explicitly reason about hand-object contact. The high-quality contact ground truth (along with RGB-D data) can help in the development, and more importantly, evaluation of hand mesh deformation algorithms.



(a)



(b)

Figure 3.10: (a) Image-based contact prediction architecture. (b) Contact prediction from RGB images, using networks trained with 3 views. Hand poses shown only for reference.

CHAPTER 4

CONTACTGRASP: FUNCTIONAL MULTI-FINGER GRASP SYNTHESIS FROM CONTACT

Abstract: Grasping and manipulating objects is an important human skill. Since most objects are designed to be manipulated by human hands, anthropomorphic hands can enable richer human-robot interaction. Desirable grasps are not only stable, but also functional: they enable post-grasp actions with the object. However, functional grasp synthesis for high degree-of-freedom anthropomorphic hands from object shape alone is challenging. We present ContactGrasp, a framework for functional grasp synthesis from object shape and contact on the object surface. Contact can be manually specified or obtained through demonstrations. Our contact representation is object-centric and allows functional grasp synthesis even for hand models different than the one used for demonstration. Using a dataset of contact demonstrations from humans grasping diverse household objects, we synthesize functional grasps for three hand models and two functional intents. The project webpage is `https://contactdb.cc.gatech.edu/contactgrasp.html`.

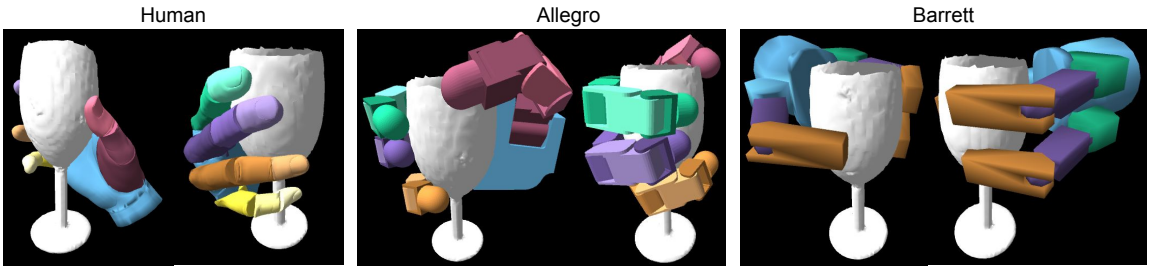


Figure 4.1: ContactGrasp synthesizes functional grasps for diverse hand models. Here we show grasps for 'using' a wine glass to drink from it. Left: 20-DOF human hand, Middle: 16-DOF Allegro hand, and Right: 4-DOF Barrett hand. Each hand has additional 6 DOFs for palm pose.

4.1 Introduction

Household objects are designed for use and manipulation by human hands. Humans excel at grasping and then performing actions with these objects. Enabling this skill for robots has the potential to unlock more productive and natural human-robot interaction. We take a step towards this by proposing ContactGrasp, a framework to synthesize functional grasps. Using object shape and contact demonstrations, ContactGrasp allows functional grasp synthesis for kinematically diverse hand models.

Recent work on robotic grasping of household objects focuses on using large amounts of data collected by trying random grasping actions, often using parallel-jaw or suction-cup end effectors [110, 111, 71, 112]. This approach generates lots of self-supervised data that enables training of robust grasp policies. However, the simplicity of the end effectors has a big hand in enabling this data-collection strategy. In addition, such end effectors are mostly suited for pick-and-place tasks and do not allow performing a post-grasp action (e.g. handing an object off, clicking the camera shutter button, switching on a flashlight, etc.). Functionality of a grasp is important because 1) it enables more natural collaboration in activities with humans, and 2) most household objects are designed with specific functional grasps in mind (e.g. spray bottle has contoured neck and squeezer, hammer has a long handle, etc.).

A large body of work addresses grasp synthesis from object geometry. These approaches model the hand as a kinematic tree of rigid mesh parts and intelligently sample its configuration space to synthesize a set of stable grasps [113, 114, 115]. However, these grasps lack the notion of functionality and most of them are far from how a human would grasp the object (see Figure 4.4 for examples).

Other approaches have utilized human demonstrations to close this gap. The human hand pose (captured by data gloves in [116], and by visual recognition in [117]) is mapped by hand-engineered transformations to a similar robot end-effector pose.

Kinesthetic teaching [118] can deliver demonstrations directly in the target hand model space. However, such kinematic re-targeting methods are tied to a specific end-effector and require careful analysis to develop mappings to new end-effectors. Approaches that record the human hand pose suffer from the additional difficulty of orienting the hand pose w.r.t. the object (which requires embedding an additional 6-DOF magnetic tracker in the object [119]). In addition, they lack a clear objective to reproduce the demonstrated contact, as we show in Section 4.6. Analysis of contact during human grasping has shown that humans prefer to contact specific areas during functional grasping [77].

This motivates the development of an object-centric approach that is not tied to a specific end-effector, and that emphasizes the reproduction of demonstrated contact. Towards this end, we propose ContactGrasp, a framework which synthesizes grasps from both object geometry and contact on the object surface. Contact can be specified manually, through human demonstrations (e.g. ContactDB [77]), or a combination of both. We show in Section 4.6 that ContactGrasp can be used to synthesize grasps that reproduce the demonstrated contact for multiple different hand models. Grasp synthesis from contact has the drawback that multiple hand configurations can sometimes result in the same contact pattern. To address this, ContactGrasp adopts a sample-and-rank approach which outputs a ranked set of grasps. We show qualitatively and quantitatively in Section 4.6 that the desired grasp can be found among the top ranked grasps in this set.

To summarize, we make the following contributions:

- Develop a multi-point **contact representation** that supports efficient grasp synthesis.
- Propose a sample-and-rank approach for **functional grasp synthesis** from object shape *and contact* that can work with multiple hand models.

4.2 Related Work

Grasp synthesis has been widely studied from many perspectives. *Analytic* approaches [120, 121, 122, 123, 124, 125, 126, 127] synthesize grasp(s) from object shape, which usually guarantee stability within their set of assumptions. These assumptions include perfect object models, rigid body approximation of the hand, Coulomb friction, simplified contact models, etc. While such approaches contribute valuable theoretical analysis of grasping, their simplifying assumptions are often quite restrictive. This makes purely analytic algorithms difficult to deploy in the real world.

In contrast, *data-driven* approaches rely on machine learning techniques to learn features in some representation of the object (RGB image, 3D mesh, etc.) that can be used to predict grasps. For example, Li et al [128] compute shape features for both objects and grasping hands, and use nearest-neighbor retrieval to synthesize grasps for novel objects. Newer deep-learning based algorithms typically require large amounts of training data, which is expensive to collect and label. In addition, they are limited to simple end effectors like parallel jaw grippers to enable efficient training data collection. For example, [129, 130, 131] predict grasps for a parallel jaw gripper from a dataset of manually annotated images. Some recent works like Pinto et al [112] have used self-supervision to automate data collection, again for a simple parallel jaw gripper. We refer the reader to [132] for an extensive survey of data-driven grasp synthesis.

Hybrid approaches use analytic criteria (e.g. [133, 68]) to sample a large number of grasps in a simulator like GraspIt! [134]. Real-world demonstrations in various forms are then used to process these samples: filter them or train a machine learning algorithm to predict success. For example, Mahler et al [110, 71, 111, 112] execute those grasps with a robot and record success/failure. Song et al [135] learn a Bayes Net to jointly model post-grasp task and discretized hand pose. Synthetic data generated

using a grasp planner is labeled manually for post-grasp task suitability. However, the algorithm is not aware of contact and the pose discretization often results in predictions that are not in contact with the object.

Grasp Synthesis from Contact: Hamer et al [24] record human demonstrations of grasping by in-hand scanning to get both object and hand pose. Contact points are aggregated on the object surface and used to form a prior for hand pose synthesis and tracking. In contrast to ContactGrasp, their algorithm requires demonstrations of both hand pose and contact. Ben Amor et al [136] learn a low-dimensional grasp space from human demonstrations acquired using a data-glove. This is used along with manually specified contact points to optimize the final robot grasp. In addition to requiring manual specification of per-finger contact point, it is not clear how well the low-dimensional space can recover the finegrained functional grasps synthesized by ContactGrasp. Varley et al [137] replace human demonstrations with synthetic data by running a grasp planner in simulation and recording the fingertip contact points. Ye et al [138] develop an algorithm to sample physically plausible contact points between the hand the object, given the wrist and object pose from a motion-capture system. These sampled points are then used to synthesize realistic-looking hand poses. These methods approximate contact as a single point per fingertip. In contrast, ContactGrasp allows realistic multi-point contact (see Figure 4.2), allows using kinematically diverse hand models, and demonstrates good performance across more complex and numerous objects.

4.3 Contact Model and Human Demonstrations

As mentioned in Section 4.1, ContactGrasp can leverage human demonstrations of grasp contact to synthesize similar grasps for various hand models. It has been shown that humans contact grasped objects with not only fingertips, but also the palm and non-tip areas of the fingers. Hence our contact model and grasp synthesis algorithm

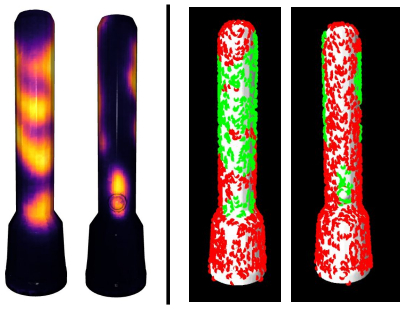


Figure 4.2: Contact map construction for the 'flashlight' object from ContactDB human demonstration. Points are randomly sampled on the object surface. green: attractive, red: repulsive.

supports multi-point contact.

We define the contact map $\mathbf{c}$ as a set of N points p_i sampled uniformly at random on the object surface, with a contact value c_i of $+1$ (attractive) or -1 (repulsive). Contacted points in the demonstration are marked attractive, while others are marked repulsive. Repulsive points allow ContactGrasp to exploit negative information [139]. Figure 4.2 shows an example of the contact map for the 'flashlight' object, with attractive points in green and repulsive points in red.

4.3.1 Human Contact Demonstrations

The contact model specified above supports manual specification. However, most of our experiments are performed using real-world contact maps from the ContactDB dataset [77]. ContactDB uses a thermal camera to observe the thermal after-prints left by heat transfer from hand to object during grasping. It is thus able to texture the object mesh surface with high-resolution contact maps. Participants grasp the objects with one of two post-grasp functional intents: *handoff* or *use*. Since ContactDB contact maps have continuous values $t(p_i) \in [0, 1]$, we threshold them at τ_t to determine whether a point is attractive or repulsive. Figure 4.2 shows an example

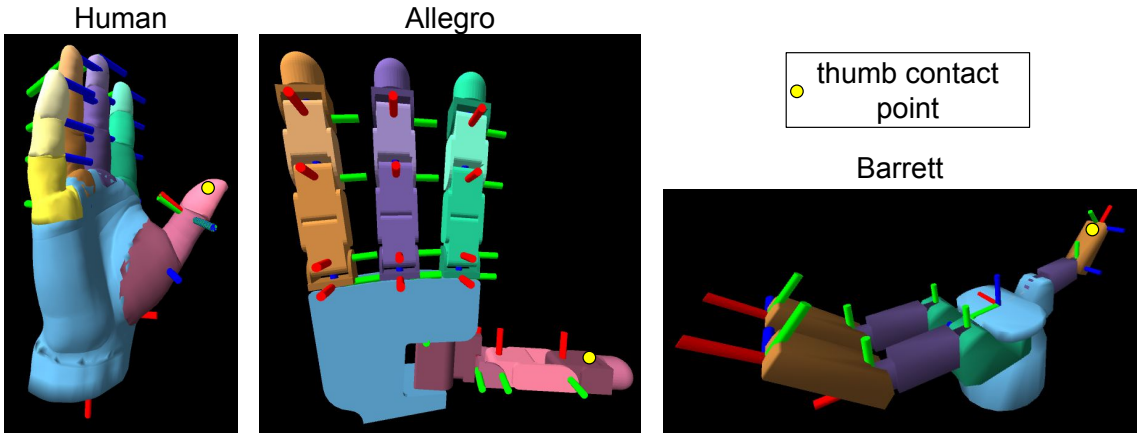


Figure 4.3: Hand models used in our experiments, along with joint axes and special thumb contact point. Left: HumanHand [134], middle: Allegro Hand [140], right: Barrett Hand [141]

of human contact demonstration from the ContactDB dataset.

$$c_i = \begin{cases} +1, & \text{if } t(p_i) \geq \tau_t \\ -1, & \text{otherwise} \end{cases} \tag{4.1}$$

4.4 Hand Models

Given the contact map on the object surface, an articulated hand model is used to set up an optimization problem which seeks a hand pose that agrees with the observed contact map. We use three kinematically diverse hand models to show that the object-centric contact representation of ContactGrasp allows grasp synthesis with diverse hand models (See Table 4.1 and Figure 4.3). Each model is a kinematic tree, with parts modelled by rigid meshes. In addition to the articulation DOFs d_i the overall position of the hand in 3D space is defined by a 6 DOF rigid body transform T. We denote the full pose of the hand as $\Phi = \left(T, \{d_i\}_{i=1}^{D}\right)$, where D is the number of articulation DOFs of the hand. We compute a signed distance field (SDF) around each hand part and attach it to the part's local coordinate system to support hand pose optimization. See Section 4.5 for further details.

Table 4.1: Hand models used in our experiments.

Model	Fingers	Joints	Articulation DOFs
HumanHand [134]	4 fingers, 1 thumb	15	20
Allegro [140]	3 fingers, 1 thumb	12	16
Barrett [141]	3 digits	7	4

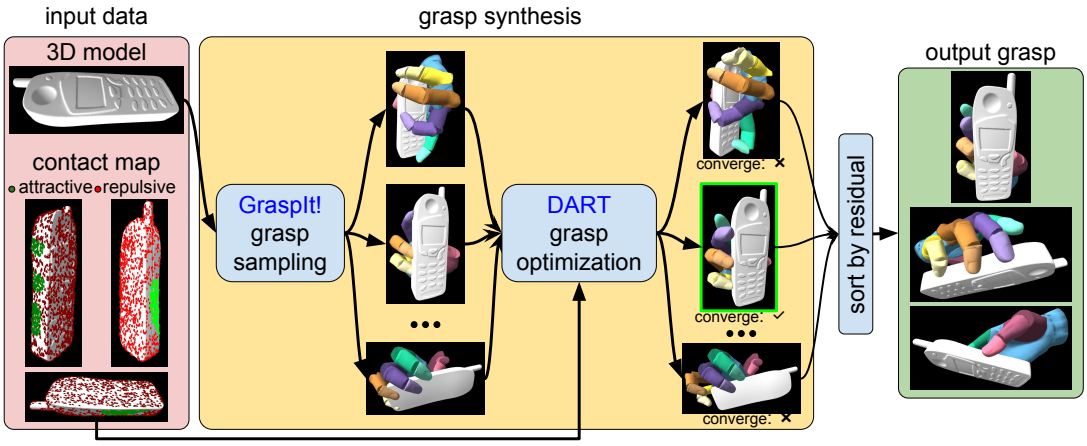


Figure 4.4: Overview of the ContactGrasp algorithm. GraspIt! [113] is used to sample random grasps for the object geometry. These are then refined and ranked for agreement with a human-demonstrated contact map to synthesize functional grasps.

4.5 ContactGrasp: Grasp Synthesis

In this section, we describe the algorithm to synthesize grasps from object geometry and a contact map. Grasp synthesis consists of estimating the full configuration Φ of the hand, which includes the 6-DOF 'palm' pose as well as joint values. This is done through an appropriately initialized nonlinear optimization. Figure 4.4 shows an overview of the entire algorithm.

4.5.1 Grasp Optimization

Intuitively, the objective of this optimization is to encourage contact at attractive points and discourage contact at repulsive points in the contact map. The full objective function consists of three terms, inspired from [142] (see Figure 4.6 for a visual example of each term):

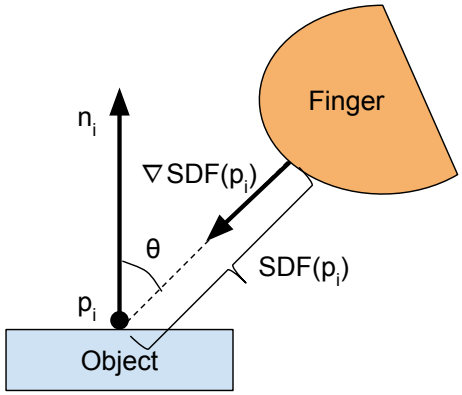


Figure 4.5: Geometry of the activation function for a repulsive point p_i in the contact map for an object.

Grasp Term

The grasp term attracts (resp. repels) the closest hand segment to every attractive (resp. repulsive) point in the contact map. For a given contact map $\mathbf{c}$ and hand pose Φ, the term is stated as:

$$L_{grasp}(\Phi|\mathbf{c}) = \sum_{i=1}^{N} \lambda_a [c_i = +1] SDF_{k_i}(p_i)^2 - \lambda_r [c_i = -1] SDF_{k_i}(p_i)^2 \tag{4.2}$$

Where k_i is the index of the hand segment that is closest to point p_i, $SDF_k(\cdot)$ is the signed distance function (SDF) associated with hand segment k, and $[\cdot]$ is the indicator function. λ_a and λ_r are hyperparameters controlling the strength of attractive and repulsive points. However, Eq. 4.2 unnecessarily penalizes some hand poses. As shown in Figure 4.5, we want to repulse a hand part away from a repulsive point p_i only if it is directly above the repulsive point i.e. the vector connecting p_i to the nearest point on the hand is parallel to the surface normal at p_i. Since SDF measures Euclidean distances, it penalizes a nearby hand part even if it is not directly above p_i. Hence we modify the activation condition for repulsive points in Eq. 4.2 as follows,

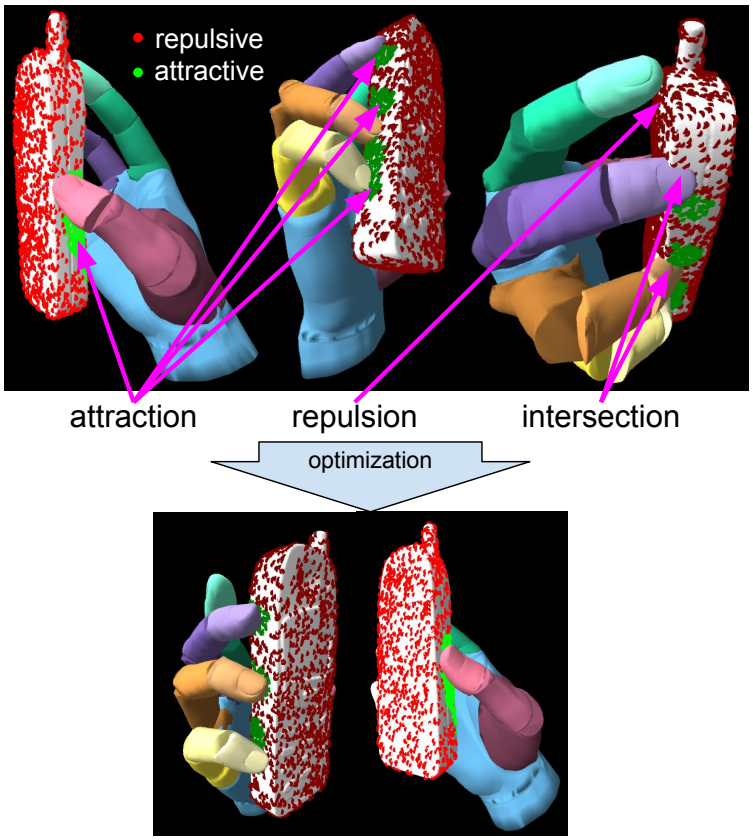


Figure 4.6: Top: Various factors involved grasp optimization. Bottom: Optimized result.

taking the surface normal n_i at point p_i into consideration (see also Figure 4.5):

$$L_{grasp}(\Phi|\mathbf{c}) = \sum_{i=1}^{N} \lambda_a[c_i = +1]SDF_{k_i}(p_i)^2 - \lambda_r[c_i = -1]f_{k_i}(p_i, n_i) \tag{4.3}$$

where

$$f_{k_i}(p_i, n_i) = \begin{cases} SDF_{k_i}(p_i)^2, & \text{if } |\widehat{\nabla SDF}_{k_i}(p_i) \cdot n_i| > \tau_n \\ 0, & \text{otherwise} \end{cases} \tag{4.4}$$

where $\widehat{\nabla SDF}(\cdot)$ is the unit vector in the direction of the gradient of the SDF.

Thumb Contact Term

It is well known that the thumb is especially important in human grasps [143]. To account for this, we specify a point p_{thumb} on the thumb (or the part closest resembling the thumb) in the hand model (yellow dots in Figure 4.3). The thumb contact term encourages p_{thumb} to be in contact with the object:

$$L_{thumb}(\Phi) = \lambda_t SDF_{object}(p_{thumb})^2 \tag{4.5}$$

where $SDF_{object}(\cdot)$ is the signed distance function associated with the object, and hyperparameter λ_t controls the strength of this term.

Intersection Term

The intersection term $L_{int}(\Phi)$ discourages intersection of the hand model with the object and self-intersection among segments of the hand, and is the same as the one used in [142]. Its strength is controlled by the hyperparameter λ_i.

Optimization

The full objective function for grasp optimization is given by

$$L(\Phi|\mathbf{c}) = L_{grasp}(\Phi|\mathbf{c}) + L_{thumb}(\Phi) + L_{int}(\Phi) \tag{4.6}$$

We use Dense Articulated Real-time Tracking (DART) [139] to minimize Eq. 4.6 and get the optimized hand pose. Specifically, we modify the Contact Prior mechanism in DART [142] to 1) support repulsive points, and 2) remove the depth-map observation term. DART approximately minimizes Eq. 4.6 by running the Levenberg-Marquadt algorithm (see [139] for more details).

4.5.2 Initializing the Grasp Optimization

Since the Levenberg-Marquadt grasp optimization is local, and search in the high-dimensional hand pose space has many local minima, providing good initialization to the optimizer becomes important. Towards this end, we develop an algorithm to sample diverse grasps for the object geometry which are agnostic to the contact map, using the publicly available GraspIt! grasp planner [134, 113]. These are later ranked for agreement with the contact map using the residue after minimizing Eq 4.6.

Specifically, we seed the planner with a coarse grasp $\phi = (\mathbf{a}, \theta, d)$, where $\mathbf{a}$ is the approach point on the object surface, θ is the roll angle around approach vector and d is the distance from object surface. We use the negated surface normal $-\mathbf{n}$ at the approach point $\mathbf{a}$ as the approach vector, making since the hand approach anti-parallel to the object surface normal (similar to [144]). Approach points $\mathbf{a}$ are sampled uniformly at random over the object surface, whereas θ and d are set from the discrete sets {0, 90, 180, 270} and {0 cm, 1 cm, 2 cm, 3 cm} respectively. GraspIt!'s Simulated Annealing planner then runs for 45K iterations for each seed to optimize the Contact Energy cost function [113], exploring in a cone around the specified approach vector.

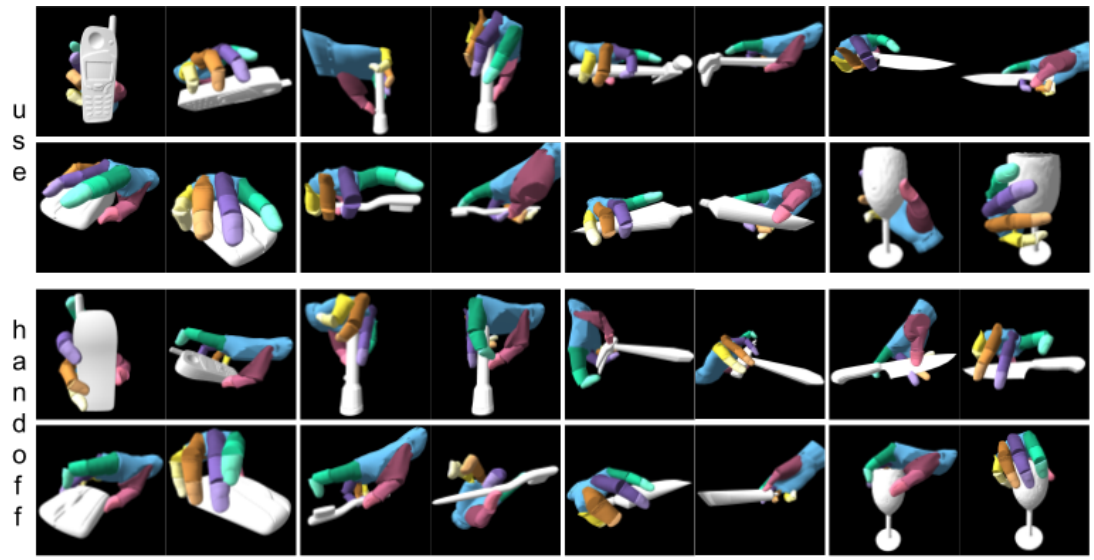


Figure 4.7: Functional HumanHand grasps synthesized by ContactGrasp. Top: Use, bottom: Hand-off.

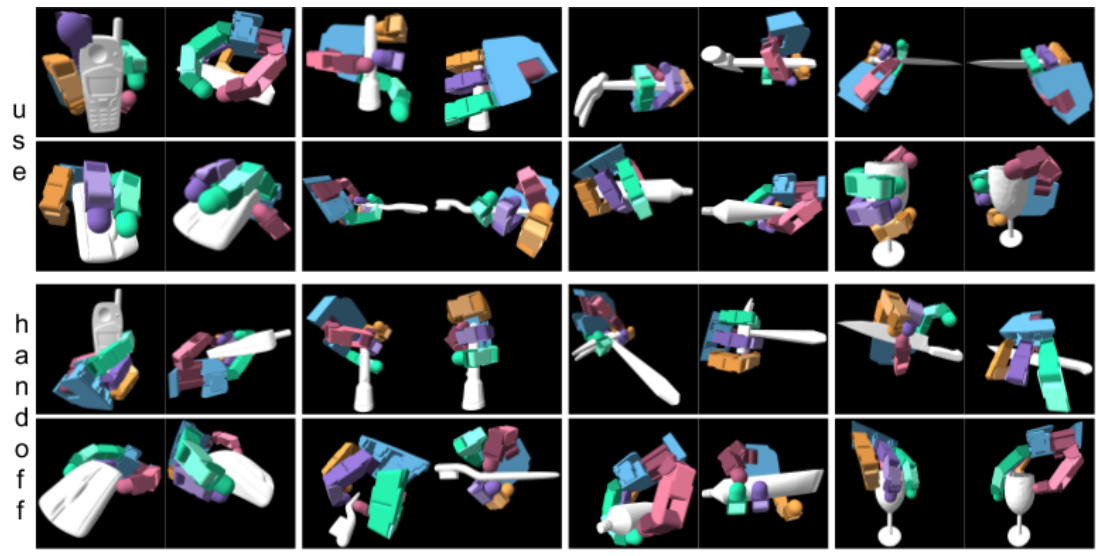


Figure 4.8: Functional Allegro hand grasps synthesized by ContactGrasp. Top: Use, bottom: Hand-off.

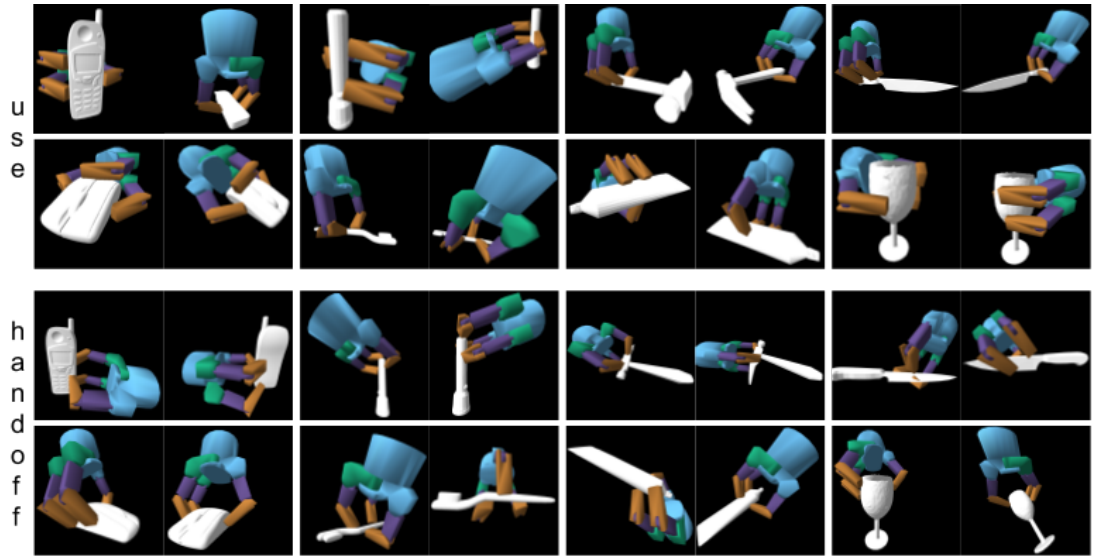


Figure 4.9: Functional Barrett hand grasps synthesized by ContactGrasp. Top: Use, bottom: Hand-off.

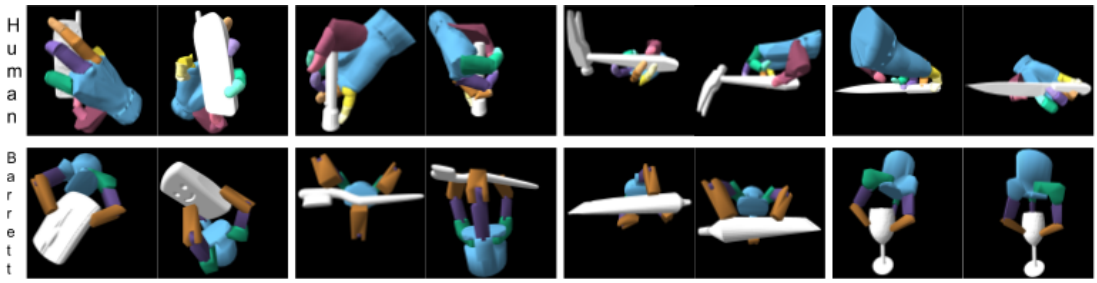


Figure 4.10: Top-ranked grasps from GraspIt! [113], which are agnostic to contact map and hence are not functional, especially for the 'use' intent. For example, flashlight button is not accessible by the thumb, finger touches knife blade, cellphone screen and wineglass opening are blocked by palm. Top: Human Hand, Bottom: Barrett Hand.

We pick the top 2 grasps after each run of the planner, and add them to the set of sampled diverse grasps $\mathcal{D}$. Note that $\mathcal{D}$ contains full grasps, since GraspIt! provides a bridge to convert coarse grasps ϕ to full grasps Φ through its planner. Figure 4.4 shows some grasps sampled in this manner for a cellphone.

The last step is to refine and rank the grasps in $\mathcal{D}$ by running grasp optimization on each of them, and considering the residual after convergence (Eq. 4.6) as the negative score. Most of the grasps in $\mathcal{D}$ result in large residuals because they are agnostic to the contact maps and hence can be out of the basin of convergence of the local optimization. Hence this ranking process causes the 'correct' hand pose to show up among the top-ranked, from which it can be easily identified.

To summarize, the process described in this section is recovers the correct full hand pose that agrees with a given contact map on an object. Manually annotating the full hand pose prohibitively expensive because of the high dimensionality.

4.6 Results

We use the ContactGrasp algorithm (Section 4.5) to synthesize grasps for a diverse set of household objects for two different post-grasp functional intents: *using* the object, and *handing it off*. Grasps are synthesized for the three hand models described in Section 4.4. We use human contact demonstrations for functional grasps from the ContactDB dataset [77]. 25 objects in ContactDB have demonstrations for both using the object and handing it off. From these, we select a subset of 19 objects (see supplementary video for a list) for which bi-manual grasps were not observed. Note that the grasp optimization strategy described in Section 4.5 supports bi-manual grasps. However, we focus on single-handed grasps here owing to lack of left-handed hand models and need for a more complex initialization strategy for the optimization.

We set the hyperparameter τ_t (threshold on contact map value) to 0.3, λ_a (attractive contact point strength) to 150.0, λ_r (repulsive contact point strength) to 20.0, λ_t

Table 4.2: **Dis**agreement of the ContactGrasp and GraspIt! grasps from human-demonstrated contact, measured by L_{grasp} (see Eq. 4.3, lower is better). Human-Allegro indicates grasps mapped from human to the Allegro model using [145].

Intent	**End-effector**	**ContactGrasp** L_{grasp}	**GraspIt!** L_{grasp}
use	HumanHand [134]	**-0.07**	0.17
	Allegro [140]	-0.06	0.32
	Human - Allegro [145]	0.08	0.12
	Barrett [141]	-0.03	0.19
handoff	HumanHand [134]	**-0.14**	0.19
	Allegro [140]	-0.11	0.41
	Human - Allegro [145]	0.05	0.15
	Barrett [141]	-0.04	0.22

(thumb contact point strength) to 25.0, and λ_i (intersection term strength) to 100.0.

4.6.1 Qualitative results

Figures 4.7, 4.8, and 4.9 show the synthesized grasps for the HumanHand, Allegro hand and Barrett hand respectively, for 8 objects and 2 functional intents(see supplementary video for other objects). They demonstrate functional grasps e.g. for the 'use' intent, the flashlight button is easily accessible by the thumb, fingers rest on mouse click buttons, knife is held by the handle. In contrast, the top-ranked grasps from GraspIt! are shown in Figure 4.10. They are stable but do not demonstrate functionality, especially for the 'use' intent e.g. flashlight button is not accessible by the thumb, finger touches knife blade, cellphone screen and wineglass opening are blocked by palm.

4.6.2 Quantitative results

Table 4.3 shows the median rank (across all objects) of the synthesized grasp, when grasps are ranked according to 1) the DART residual, and 2) GraspIt!'s Contact Energy metric [113]. The median rank is significantly lower for the former. This indicates that approaches like [113] that consider only object geometry cannot guar-

Table 4.3: Median rank of the correct grasp (lower is better).

Intent	End-effector	ContactGrasp rank	GraspIt! rank
use	HumanHand [134]	**3(0.03%)**	4484(34.41%)
	Allegro [140]	22(0.69%)	1289(40.28%)
	Barrett [141]	6(0.19%)	2362(74.09%)
handoff	HumanHand [134]	**1(0.01%)**	3751(38.37%)
	Allegro [140]	22(0.56%)	1258(38.20%)
	Barrett [141]	24(0.75%)	830(25.94%)

antee functional contact. ContactGrasp can effectively leverage these approaches to synthesize functional grasps using demonstrations.

Table 4.2 shows further quantitative results in the form of the L_{grasp} value (see Eq. 4.3) for the synthesized grasp as well as the top-ranked grasp by GraspIt!'s Contact Energy metric. L_{grasp} measures the grasp's disagreement with the demonstrated contact map, and is significantly lower for ContactGrasp grasps. This verifies that ContactGrasp produces grasps that are significantly closer to the human demonstrations than GraspIt!, which is agnostic to the demonstrations.

In addition, we implement the kinematic retargeting algorithm of Tosun et at [145] for mapping the synthesized human hand grasp to the Allegro model. Each finger is treated as a separate kinematic chain (sampled with 50 points) to be re-targeted. The human pinky finger is discarded. The L_{grasp} values for these mapped grasps are higher than those for the ContactGrasp Allegro grasps as well as human grasps. This shows that deterministic mapping of grasps between hand models does not reliably reproduce contact, supporting our motivation for developing ContactGrasp.

4.6.3 Failure Cases

Figure 4.11 shows failure cases of ContactGrasp. These occur when the grasp sampling stage is not exhaustive enough to sample fine manipulation behaviors like fingers through holes, or when the target hand model does not possess the geometry to be able to achieve a complicated control pattern.

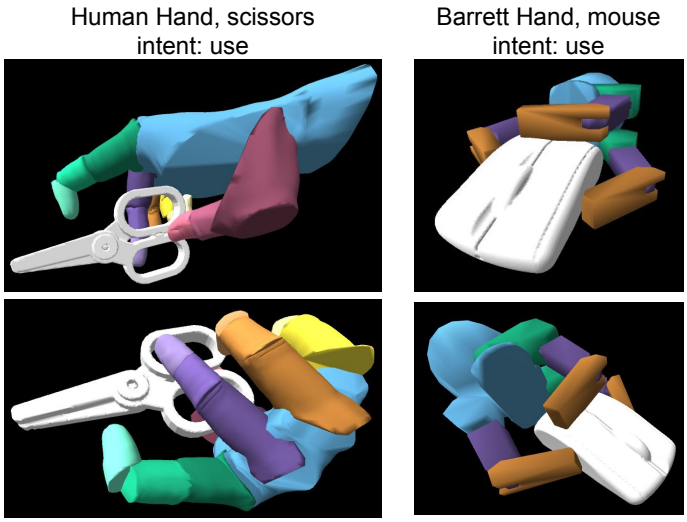


Figure 4.11: Failure cases. Left: Grasp sampling is unlikely to produce a grasp with fingers in narrow holes, and optimization then gets stuck in local minima. Right: Some end effectors lack the structure for functional grasps e.g. resting fingers on mouse buttons.

4.7 Conclusion

To summarize, we develop a multi-point contact model, which can plug-and-play with kinematically diverse hand models to synthesize grasps. These grasps can be modulated by hand-object contact demonstrations to be functional, supporting post-grasp actions like using the object or handing it off. We show that our approach ContactGrasp, which directly optimizes for contact, is superior than other approaches which kinematically re-target observed human grasps to the target hand model. We demonstrate the effectiveness of ContactGrasp by synthesizing functional grasps for 3 significantly diverse hand models, 19 household objects, and 2 functional intents.

CHAPTER 5
TOWARDS PREDICTION OF CONTACT PRESSURE FROM CONTACT MAPS

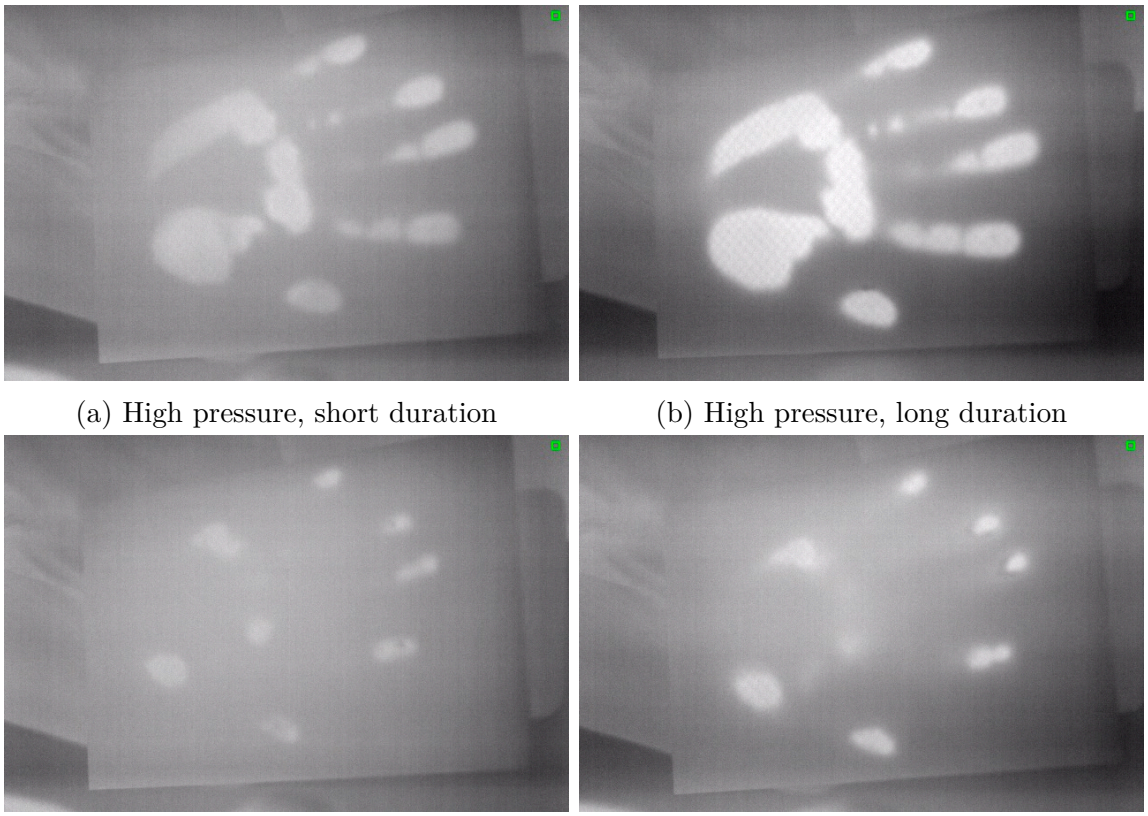
5.1 Introduction

Previous chapters used contact maps to capture the *location* of contact between the hand and the object. Contact locations, however, only partially describe a grasp. Forces exerted by all parts of the hand complete the description. This chapter explores whether contact maps also capture contact *pressure* in addition to contact location.

Some factors that might influence grasp contact forces are: grasp function (power grasps typically use more force than precision grasps [146]), object weight (heavier objects are grasped more firmly to prevent slip [147, 148]), and object material (delicate objects are usually grasped with less force). Pressures can be calculated by dividing contact forces by contact areas.

Clues about the precise values and distribution of contact pressure are difficult to observe through RGB and/or depth images (unless the camera has a very high resolution or is very near the hand, in which case it can see changes in fingernail and skin blood flow). However, we noticed while capturing contact maps with thermal cameras, that contact pressure influences not only the intensity of contact maps, but also their structure. Figure 5.1 shows an example. Notice how higher contact pressure significantly alters the contact map by bringing more surface area under contact through soft tissue deformation. In contrast, longer contact duration only increases the contact map intensity (presumably because of more heat transfer).

This chapter investigates if this effect can be used to infer contact pressure from the structure of the contact map. We focus on instances of contact with a planar surface.



(a) High pressure, short duration

(b) High pressure, long duration

(c) Low pressure, short duration

(d) Low pressure, long duration

Figure 5.1: Effect of contact pressure and duration on contact map structure and intensity.

This allows the use of a high-resolution pressure sensor (which is only available in a planar form factor) to record pressure values for quantitive evaluation, and removes the influence of object shape on soft tissue deformation. To summarize, we make the following contributions:

- **Data**: We create a new dataset of paired RGB, thermal, and pressure images depicting hands contacting a planar surface with various levels and distributions of pressure. This dataset, named ContactPressure, will be released publicly for research use.
- **Pressure prediction models**: We evaluate various representations of hand pose and pressure for contact pressure prediction from thermal contact maps.

5.2 Related Work

Image- and video-based contact force prediction has prevously been studied in the hand pose estimation and tracking context. Pham *et al.*[80, 19] track 3D models of the hand and object in manipulation videos, and predict contact forces at fingertips that explain the observed motion. They also learn the distribution of force among fingertips by recording pressure data from force sensors embedded in a cuboidal object. Ehsani *et al.*[149] collect a dataset of manipulation videos of 8 objects annotated with object keypoints and fingertip contact locations, and learn to predict contact forces in a similar manner. Contact force has also been estimated from the distribution of blood color in fingernails, by developing special sensors [150] and modeling the interaction of bone and soft tissue [151].

These approaches estimate contact forces only at pre-selected points on the hand (fingertips) because of the difficulty of annotating whole-hand contact areas, and focus on estimating 'live' contact in videos. In contrast, we investigate the estimation of contact pressure for the entire surface of hand-object contact. This is done from the

thermal contact map which has high resolution but is only available post-grasp. In addition, thermal contact maps aggregate information for the entire contact duration *i.e.* they do not allow distinguishing force changes that happen during in-hand manipulation. Hence, the post-grasp nature of our analysis trades off spatial resolution against temporal resolution. High spatial resolution of contact pressure is potentially useful for learning to imitate human grasping behavior with anthropomorphic robot hands. Currently, our algorithm has been trained and tested on planar contact maps because of the planar form-factor of pressure sensors. Unwrapping contact maps of 3D objects (*e.g.* ContactPose data) such that the corresponding hand pose resembles a flat hand, and applying these planar pressure estimation models, is an interesting direction for future work.

5.3 The ContactPressure Dataset

5.3.1 Equipment and Protocol

Figure 5.2 shows the hardware setup. A FLIR Boson 640 thermal camera is mounted on a tripod rigidly w.r.t. a Logitech C920 RGB camera. The former captures the thermal contact maps, while the latter is used to capture an RGB image of the time of contact, which is used to get hand pose information. Both cameras look at a Sensel Morph pressure sensor placed rigidly on a table. This sensor outputs 185×105 resolution 'pressure images' at 60 Hz.

We invite participants to our laboratory for data collection and use the following IRB-approved protocol for data collection. Participants use either hand to contact the Morph in a flat-hand pose for a fixed duration and with an arbitrary pressure profile (level and distribution). They are instructed to keep the pressure profile constant for the duration of contact. This is repeated at least 20 times each for 2 s, 4 s, and 6 s contact durations. The RGB camera records an image at the middle of each duration, while the Morph pressure data is max-accumulated during the entire duration. After

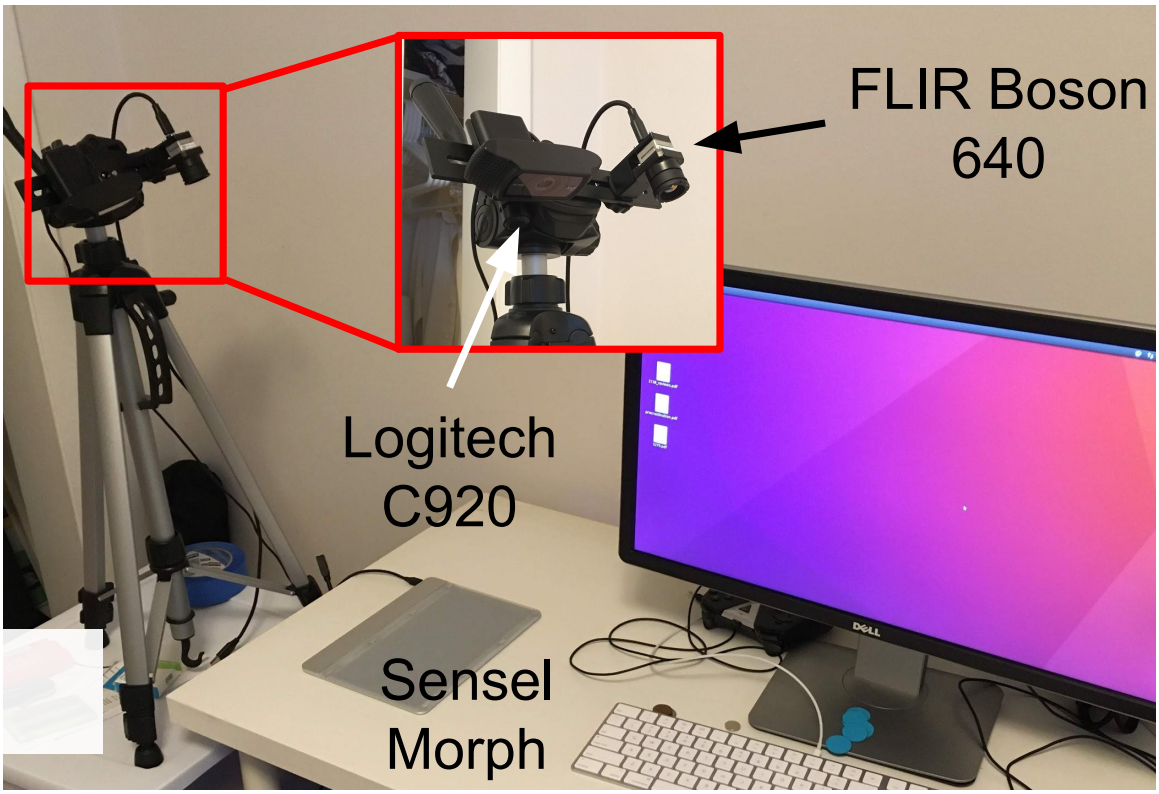


Figure 5.2: Our hardware setup consists of a Logitech C920 RGB camera and a FLIR Boson 640 thermal camera rigidly mounted on a tripod and looking at a Sensel Morph pressure sensor.

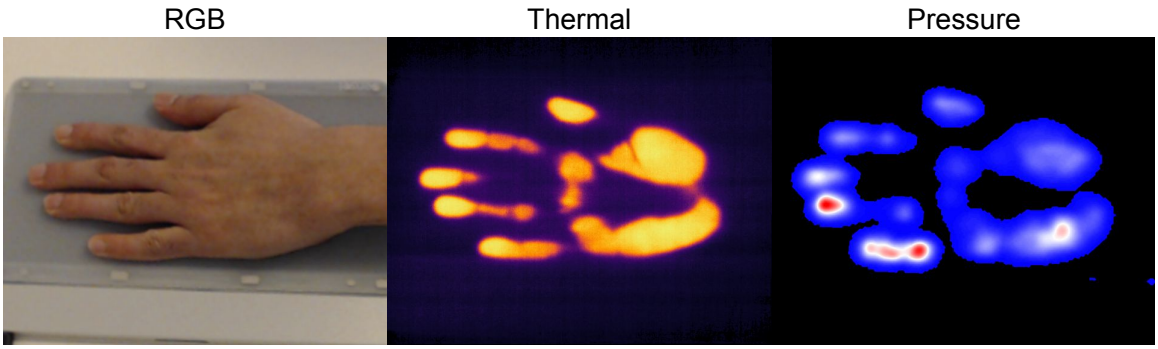


Figure 5.3: An example of registered RGB, thermal, and pressure data from the ContactPressure dataset. The thermal image color coding is similar to ContactDB (Figure 2.1) and ContactPose (Figure 3.1). The pressure image color coding goes from **blue** (low) to **red** (high).

this duration, the participants remove their hand from the camera fields of view, and the thermal camera records an image *i.e.* thermal contact map. Figure 5.3 shows an example.

5.3.2 Camera Calibration

The aim of calibration is to register images from the three modalities – RGB, thermal, and pressure. We first calibrate the intrinsics parameters of the RGB and thermal cameras. This is done using a checkerboard and a grid of circles cut out from a cardboard piece placed over a heated cardboard piece, respectively. The instrinsics parameters are used to undistort the images.

Next, we fit homography matrices connecting the RGB and thermal, and pressure and thermal images. This sufficient to register the images, since the scene (Sensel Morph surface) is planar [152]. This requires a set of corresponding points in the undistorted images. For the (RGB, thermal) pair, this is done using unquely colored paper-thin discs, which are also heated to be distinguishable in the thermal images (see Figure 5.4a). For the (pressure, thermal) pair, this is done using a heated pencil-tip eraser pressed at various locations on the Sensel Morph, which elicits small circular imprints in both modalities. Standard ellipse detection is used to detect circle centers,

Table 5.1: Breakdown of the ContactPressure dataset.

Contact Type	Participant 1		Participant 2	
	Left Hand	Right Hand	Left Hand	Right Hand
Full Palm	5	1	1	1
Partial Palm	1	1	1	1

which provides point correspondences.

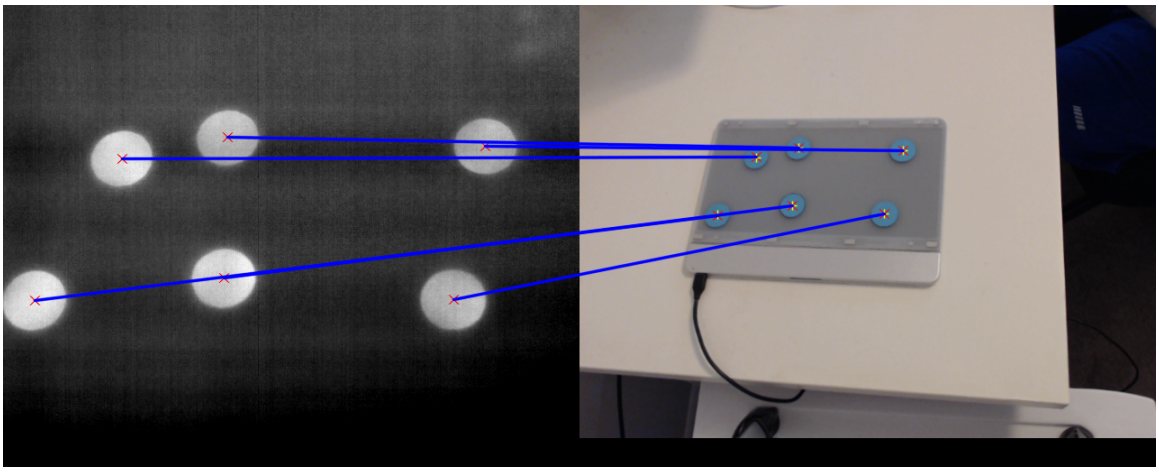
5.3.3 Dataset Scope

Currently the ContactPressure dataset has 12 sessions, with each session containing 60 RGB, pressure and thermal triplets. These 60 triplets consist of three groups (containing 20 triplets each) of 2 s, 4 s, and 6 s contact dration. The hand side and contact type are distributed as shown in the Table 5.1. 8 of these sessions were captured with one participant, while 4 were captured with another participant. In total, it has over 720 triplets.

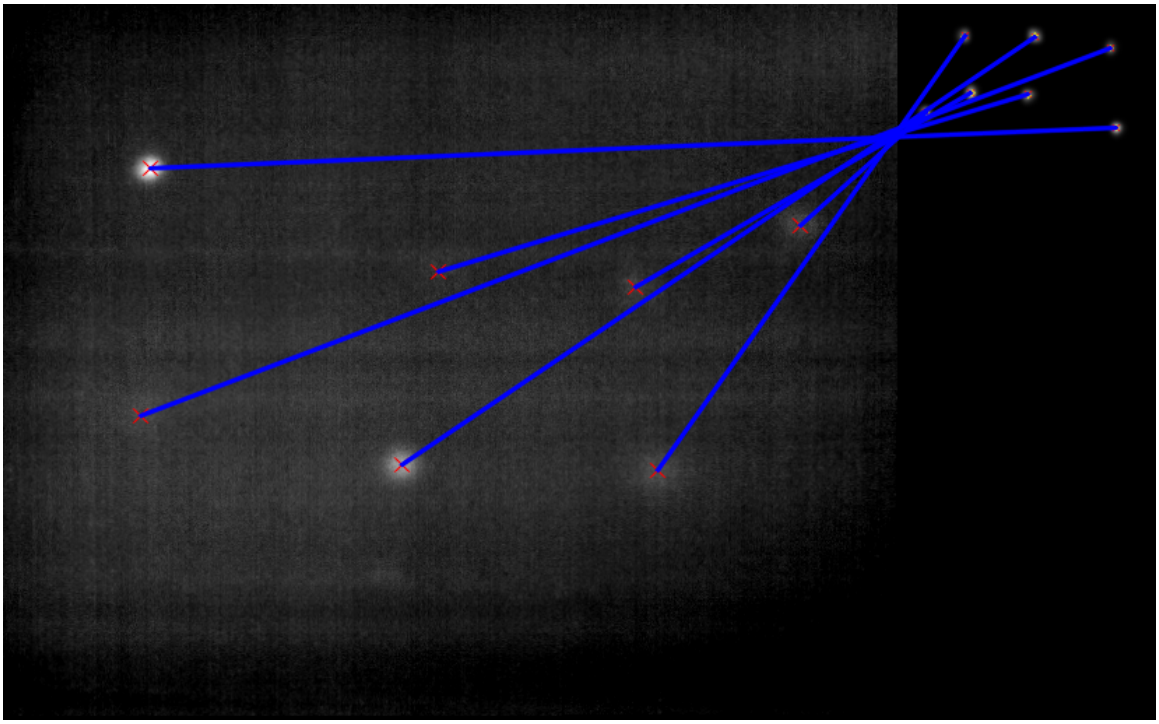
5.4 Pressure Prediction Experiments

This section describes experiments address prediction of contact pressure from thermal contact maps, and auxiliary hand pose information. These experiments focus on pressure and hand pose representation.

The calibration information described in Section 5.3.2 is used to register RGB, thermal, and pressure images in a common coordinate system. Thermal images are normalized on a per-image basis to negate the effect of raw thermal image pixel values shifting because of different hand temperatures in different sessions. As Figure 5.1 shows, pressure is related much more strongly to the structure of the contact map and intra-contact-map distribution of intensities, rather than absolute intensities.



(a) Thermal-RGB



(b) Thermal-Pressure

Figure 5.4: Detection of corresponding points for fitting homography matrices. This figure shows corresponding points through × connected by **blue** lines, and points predicted by the fitted homographies as +. (a) For the (RGB, thermal) pair this is done by placing heated teal-colored paper-thin discs at random locations on the Sensel Morph surface. (b) For the (pressure, thermal) pair this is done by pressing a heated pencil-tip eraser at random locations on the Senel Morph surface.

5.4.1 Pressure Representation

The simplest representation for pressure is to treat it as a continuous value and regress it with a mean squared-error loss. However, our initial experiments with this representation resulted in blurred and saturated predictions (similar to our observations while predicting contact in ContactPose (Chapter 3)). Similarly to contact values in ContactPose, pressure values in ContactPressure have a highly skewed distribution favoring 0 or low pressure. Hence, we employ the quantization strategy proposed by Zhang *et al.*[102] and used by us for contact modeling in the ContactPose project. Specifically, we quantize pressure values into 10 bins and treat pressure prediction as a classification problem. The cross entropy loss each bin is weighted by a value proportional to the linear combination of inverse occurrence frequency of that bin in the training dataset, and a uniform distribution (Eq. 4 from [102] with $\lambda = 0.25$). At test time we use the pixel classification scores output by our algorithm to derive a point estimate of the pressure value using annealed mean [102] with $T = 0.1$.

5.4.2 Auxiliary Information (Hand Pose) Representation

We also investigated whether information from the RGB image can be used in addition to the thermal image to improve pressure prediction. Specifically, information about the hand pose can potentially help the algorithm to learn correlation patterns between thermal contact patterns and pressure that are specific to different hand parts.

We use the publicly available OpenPose library [59, 153] to detect 2D keypoints in the RGB image (see Figure 5.5). These keypoints are then used to construct two different hand pose images of the same size as the registered thermal and pressure images:

- **`hand-pose-color`**: A unique color is associated with each phalange (line segment connecting consecutive joints). The color $\mathbf{c_i}$ for the i'th pixel in the image

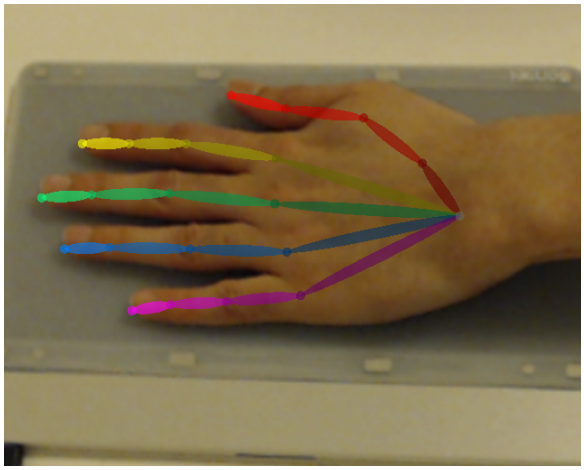
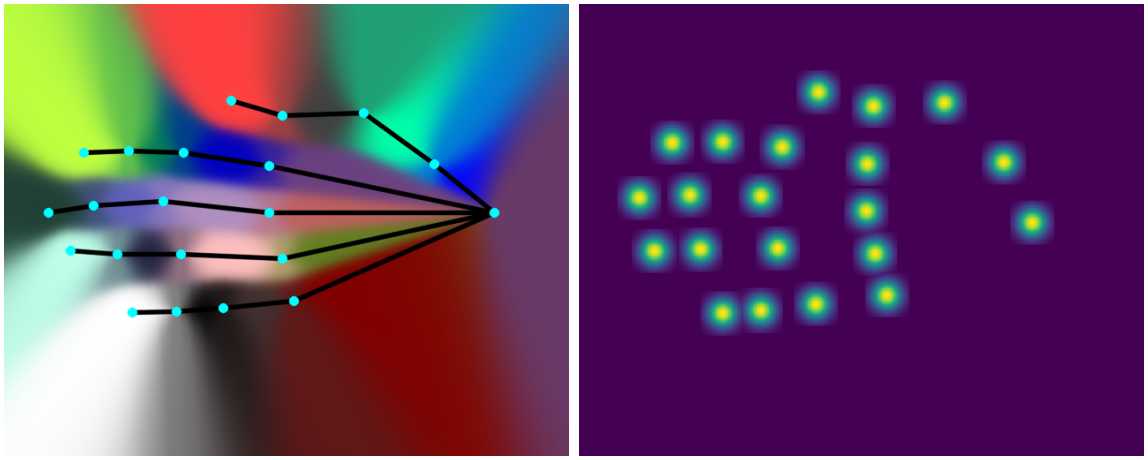


Figure 5.5: Detection of 2D joint locations in RGB images with OpenPose [153].



(a) **`hand-pose-color`** (b) **`hand-pose-joints`**

Figure 5.6: Two different representations of auxiliary hand pose information. (a) **`hand-pose-color`** color-codes the distance of each image pixel from all 20 hand phalanges. Hand joints and phalanges (line segments connecting them) are shown only for reference. (b) **`hand-pose-joints`** Places a Gaussian mass centered at each joint location to construct a 21-channel image. Here we show one such image collapsed to a single channel by applying the max operator.

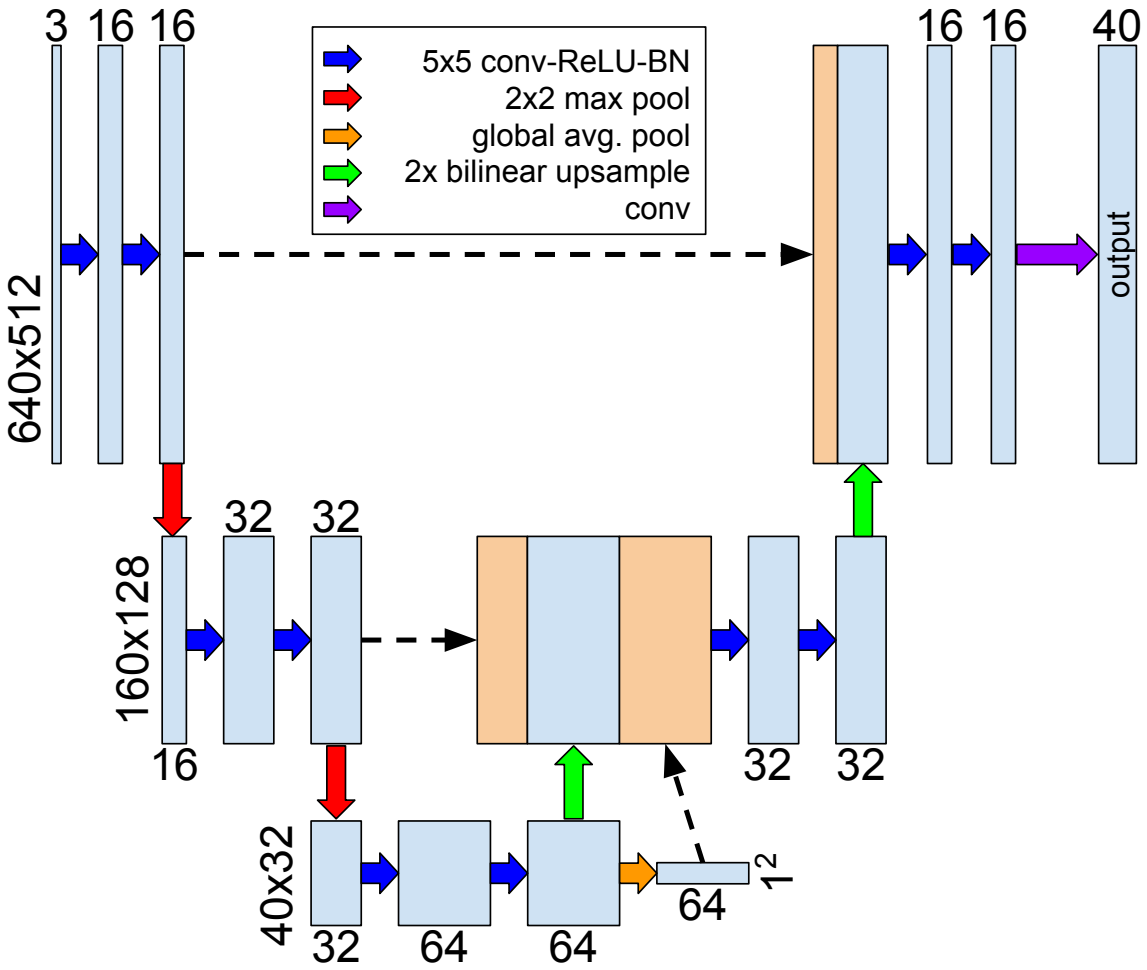


Figure 5.7: Architecture for the image encoder-decoder. Horizontal numbers indicate number of channels, and vertical numbers indicate spatial dimensions.

is a weighted linear combination of these colors, where weights w_{ij} are inversely proportional to distance of i'th pixel from the j'th phalange d_{ij}: $\mathbf{c_i} = \sum_{j=1}^{20} w_{ij}\mathbf{c_j}$, where $w_{ij} = \frac{exp(-d_{ij}/T)}{\sum_{j'=1}^{20} exp(-d_{ij'}/T)}$, $T = 7.5$. See Figure 5.6a for an example.

- **`hand-pose-joints`**: Inspired from [153], this is a 21-channel image, where each channel corresponds to a different hand joint, and has a Gaussian mass centered at the joint location. We use a 50-pixel Gaussian kernel with $\sigma = 11.1$. See Figure 5.6b for an example.

5.4.3 Convolutional Neural Network Architecture

Intuitively, the pressure value at a pixel depends on not only the local neighborhood in the correspoonding thermal (and hand pose) image, but also global information extracted from a larger support area. For example, increasing pressure at the metacarpals brings some areas of the proximal and intermediate phalanges into contact, thereby causing a long-range change in the thermal contact map.

Convolutional neural networks (CNNs) have shown the ability to learn such combinations of local and global features well, *e.g.* for semantic segmentation [154, 155]. Specifically, we use an encoder-decoder architecture with skip connections between layers in the encoder and decoder, inspired by the UNet architecture [96]. This architecture is explicitly designed to learn a combination of local and global features. Figure 5.7 shows the details.

5.4.4 Implementation Details

We trained the CNN with PyTorch [104]. Data augmentation is performed by random left-right flipping, random translation within 10% of image size, ±20 degree in-plane rotation, and randomly erasing (setting to mean pixel value) rectangular regions of the thermal and pressure images (similar to the Random Erasing augmentation proposed in [156]. The network is trained for approximately 400 epochs using the Adam optimizer [157] with an initial learning rate of 1e-3 and momentum of 0.9. Regularization is performed with a 5e-4 weight decay factor, and dropout with 0.25 probability applied to the features in the bottleneck (smallest) layer.

5.5 Pressure Prediction Results

This section discusses the results of our experiments on contact pressure prediction from thermal contact maps and auxiliary information in the form of hand pose. These

Table 5.2: Contact pressure prediction re-balanced AuC (%) (higher is better).

Input	Contact Type	Same Participant		Diff. Participant	
		Left Hand	Right Hand	Left Hand	Right Hand
Thermal Image	Full Palm	81.68	63.31	44.54	40.47
	Partial Palm	73.05	52.70	54.96	40.87
Thermal + **hand-pose-color**	Full Palm	81.13	61.99	42.48	37.56
	Partial Palm	75.40	53.91	54.71	40.31
Thermal + **hand-pose-joints**	Full Palm	**84.19**	64.72	45.82	39.62
	Partial Palm	78.49	62.28	56.56	49.23

experiments use 4 sessions of data collected from Participant 1, with full contact on the left hand. The trained models are tested on progressively more different sessions of held-out data.

For quantitative evaluation, we follow [102] and report the area under the curve (AuC) of prediction correctness threshold vs. prediction accuracy. This value is re-balanced to account for varying occurrence frequencies of values in the 10 pressure bins. Table 5.2 shows the re-balanced AuC values, and Figure 5.8 shows qualitative examples. As expected, the performance is best when the distribution of testing data is similar to training data (same participant, left hand, full palm contact) and reduces progressively with the distribution gap. We observe that auxiliary hand pose features consistently improve performance, with **hand-pose-joints** features performing better than **hand-pose-color** features. In addition, they enable the algorithm to generalize better from full palm contact to partial palm contact, but not from left hand to right hand or from one participant to another. A model trained with 'short' and 'long' duration data from all 12 sessions performed at 69.55% when tested on 'medium' duration data. Using hand shape (e.g. MANO [76] parameters) calculated from an RGB image as an additional input to enable better generalization to unseen participants out of the scope of this work, but would be an interesting direction for future research.

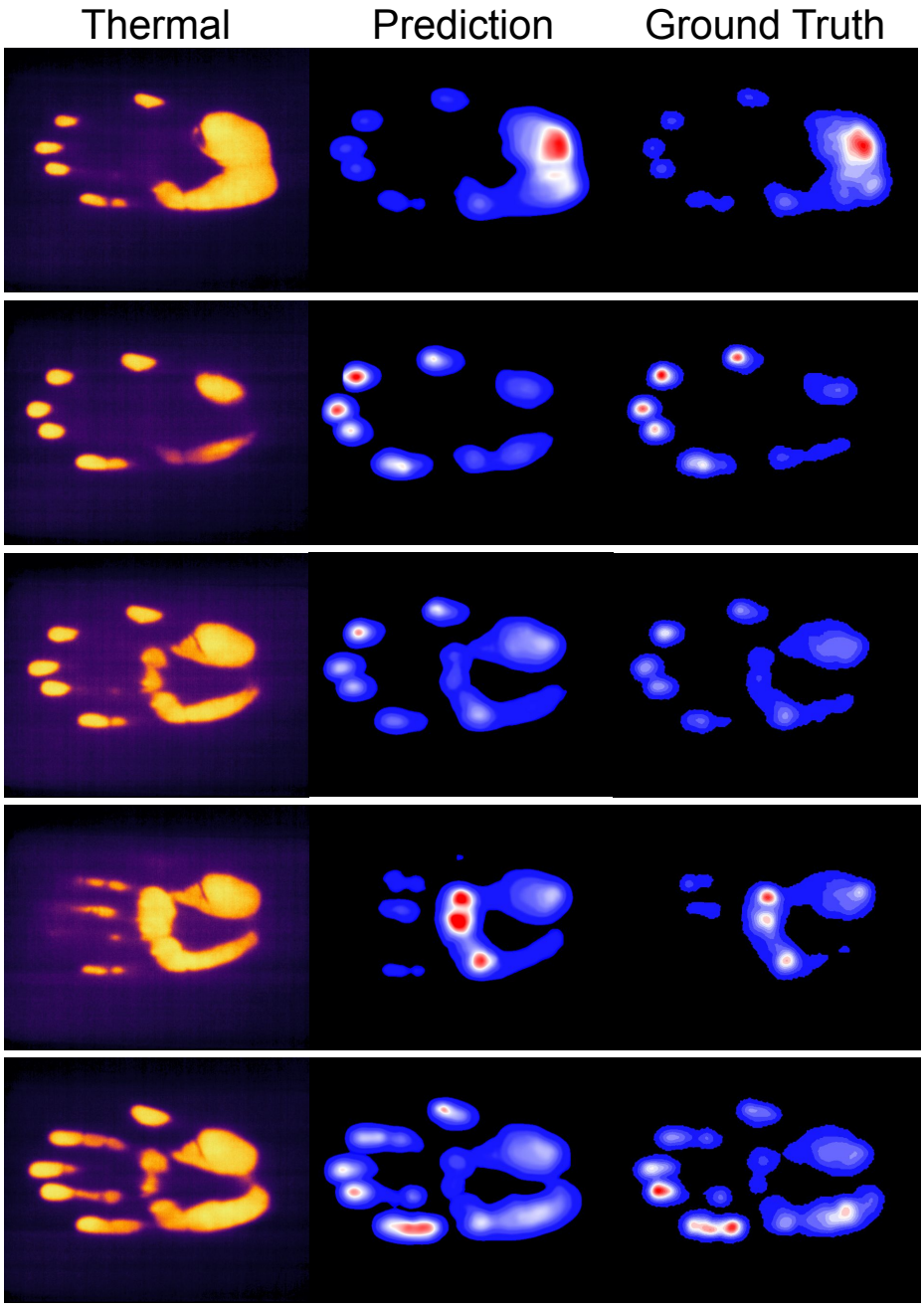


Figure 5.8: Qualiative examples of contact pressure prediction from full palm, left hand and same participant thermal contact maps, with **`hand-pose-joints`** auxiliary features. The color-coding for pressure images goes from **blue** (low) to **red** (high).

5.6 Conclusion

This chapter investigated whether contact maps obtained from thermal cameras encode information about the pressure that was applied during contact. We find that this is indeed the case, and verified that machine learning models can predict contact pressure for contact on planar surfaces. Developing algorithms that are able to generalize better to unseen hand shapes by conditioning on shape parameters, and predicting pressure for non-planar contact surfaces are intereting avenues of future research.

Appendices

APPENDIX A
CONTACTDB: ANALYZING AND PREDICTING GRASP CONTACT VIA THERMAL IMAGING – SUPPLEMENTARY MATERIAL

Abstract: This appendix provides supplementary material Chapter 2. We compare ContactDB heatmaps qualitatively against the crowdsourced tactile saliency maps from [22]. We discuss the extent of heat dissipation while scanning the object, and potential sources of error in observing contact through the thermal camera and the texture mapping process. Lastly, we list the 50 objects used in ContactDB and the instructions given to participants for grasping the subset of 27 objects with the 'use' post-grasp intent. ContactDB can be explored interactively at `https://contactdb.cc.gatech.edu`.

A.1 Comparison to Tactile Mesh Saliency [22]

Qualitatively, the closest work to ContactDB that we've found is [22], which collects contact saliency information through crowd-sourcing by pairwise comparison of surface points. Figure A.1(b) compares common objects from both datasets. Notably, data from [22] lacks clear finger-marks and resembles averaged contact maps. That data may be less accurate because it relies on self-reporting. For example, our data shows that people rarely contact the bottom half of the wine glass stem, whereas [22] shows high saliency for the entire stem.

A.2 Heat Dissipation During Data Collection

Scanning takes 18 s for a 360° rotation. Owing to the consistent use of hand-warmers and PLA material for 3D printed objects, thermal prints take more than 35 s to

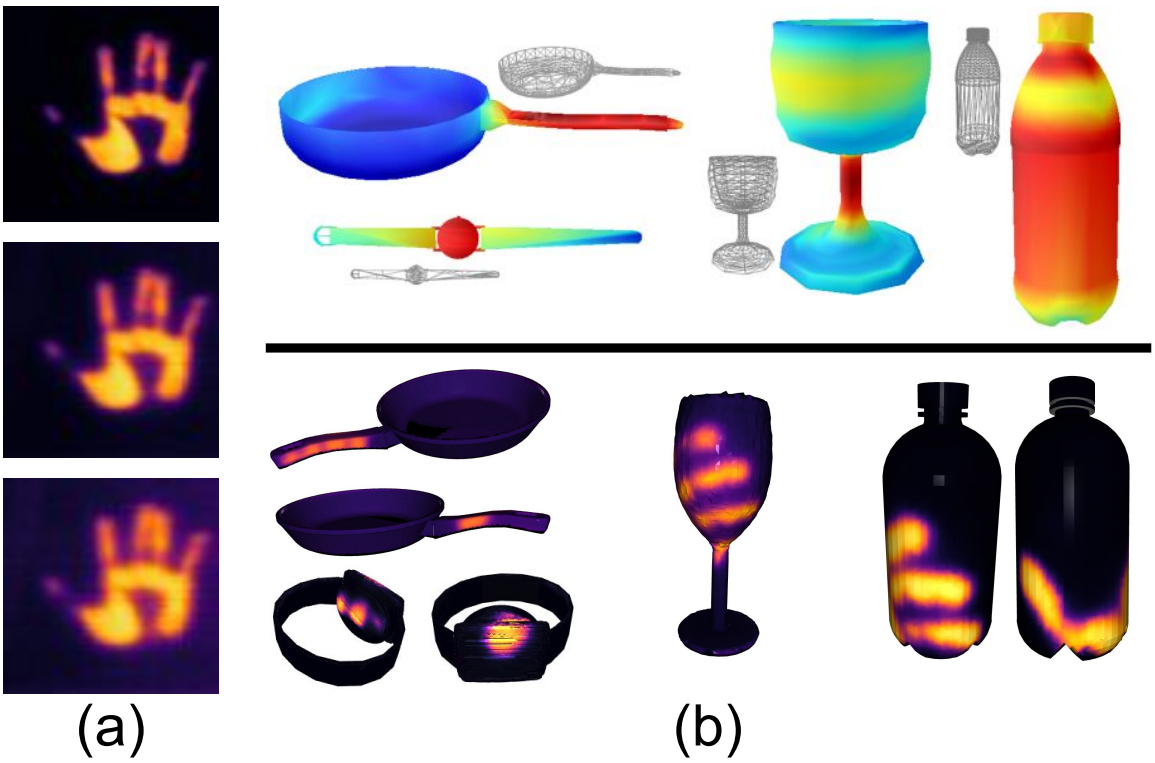


Figure A.1: (a) Heat dissipation in the thermal images. Top-bottom: 0s, 18s, 35s. (b) Contact information collected by online crowd-sourcing ([22], top row) and ContactDB (ours, bottom row).

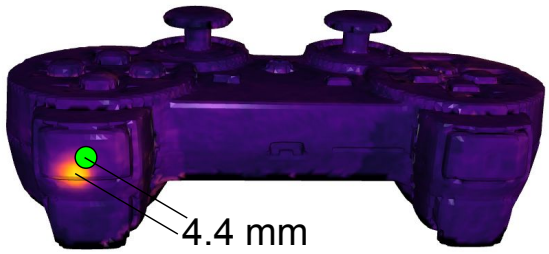


Figure A.2: Geometric error of the texture mapping process. The spot on the front button shown in green was precision-heated with a warm pencil-top eraser.

diffuse significantly (See Fig. A.1(a)). Heat conduction across the surface of the plate does not seem to be a significant source of variation between 0 s and 18 s, since the prints are comparable in size and lack strongly blurred edges. This shows that the dissipation of finger heat on the object surface produces minimal artifacts in the contact maps presented in Chapter 2. We operate the turntable motor at the maximum possible speed that avoids high centrifugal force and wear-and-tear.

A.3 Accuracy of Texture Mapping

As discussed in Section 2.3.3, thermal images from 9 views and corresponding object pose estimates are used in a texture mapping algorithm to produce a final mesh textured with a contact map. The whole process has multiple potential sources of error: calibration of the intrinsics and extrinsics of the Kinect v2 and thermal camera, inaccuracy in 3D printing the object, errors in object pose estimates due to noise/distortion in the Kinect depth maps, artifacts introduced by the texture mapping algorithm, etc. As such, the accuracy of this process can be different for different objects and sessions. In Figure A.2, we attempt to quantify this error for one instance where we precisely heated a spot on the front button of the PS controller using a heated pencil-top eraser. In this case, we observed a final geometric error of 4.4 mm.

List of Objects

Table A.1 shows a list of all 50 objects in ContactDB, along with information about the which of these objects are included in the two functional grasping categories, and the specific 'use' instructions.

Table A.1: List of objects in ContactDB and specific 'use' instructions

Object	**handoff**	**use**	**use instruction**
airplane	✓		
alarm clock	✓		
apple	✓	✓	eat
banana	✓	✓	peel
binoculars	✓	✓	see through
bowl	✓	✓	drink from
camera	✓	✓	take picture
cell phone	✓	✓	talk on
cube (small)	✓		
cube (medium)	✓		
cube (large)	✓		
cup	✓	✓	drink from
cylinder (small)	✓		
cylinder (medium)	✓		
cylinder (large)	✓		
door knob		✓	twist to open door
elephant	✓		
eyeglasses	✓	✓	wear
flashlight	✓	✓	turn on
flute	✓	✓	play
hammer	✓	✓	hit a nail
hand		✓	shake
headphones	✓	✓	wear
knife	✓	✓	cut
light bulb	✓	✓	screw in a socket
mouse	✓	✓	use to point and click
mug	✓	✓	drink from
pan	✓	✓	cook in
piggy bank	✓		
PS controller	✓	✓	play a game with
pyramid (small)	✓		
pyramid (medium)	✓		
pyramid (large)	✓		
rubber duck	✓		
scissors	✓	✓	cut with
sphere (small)	✓		
sphere (medium)	✓		
sphere (large)	✓		
Stanford bunny	✓		
stapler	✓	✓	staple
toothbrush	✓	✓	brush teeth
toothpaste	✓	✓	squeeze out toothpaste
torus (small)	✓		
torus (medium)	✓		
torus (large)	✓		
train	✓		
Utah teapot	✓	✓	pour tea from
water bottle	✓	✓	open
wine glass	✓	✓	drink wine from
wristwatch	✓		
Total	**48**	**27**	

APPENDIX B

CONTACTPOSE: A DATASET OF GRASPS WITH OBJECT CONTACT AND HAND POSE – SUPPLEMENTARY MATERIAL

Abstract: This appendix includes network architectures and training and evaluation details for various learning algorithms presented in Chapter 3, along with MANO hand mesh fitting details. It also includes examples of the RGB-D imagery present in the ContactPose dataset along with 3D hand joints projected into those images. Next, we present slices through the data in the form of 1) object- and intent-specific hand contact probabilities, and 2) 'use' vs. 'hand-off' contact maps and hand poses for some grasps of an object. Finally, we present the list of objects and their 'use' instructions, and describe the participants' hand information that is included in ContactPose.

B.1 Network Architectures

B.1.1 PointNet++

The PointNet++ architecture we use is similar to the pointcloud segmentation network from Qi et al [103], with modifications aiming to reduce the number of learnable parameters. Similarly to [103], we use $SA\left(s, r, [l_1, \ldots, l_d]\right)$ to indicate a Set Abstraction layer with a farthest point sampling ratio s, ball radius r (the pointcloud is normalized to lie in the $[-0.5, 0.5]$ cube) and d fully connected layers of size $l_i (i = 1 \ldots d)$. The global Set Abstraction layer is denoted without farthest point sampling ratio and ball radius. $FP\left(K, [l_1, \ldots, l_d]\right)$ indicates a Feature Propagation layer with K nearest neighbors and d fully connected layers of size $l_i (i = 1 \ldots d)$. $FC\left(S_{in}, S_{out}\right)$ indicates a fully connected layer of output size S_{out} applied separately to each point (which has S_{in}-dimensional features). Each fully connected layer in the Set Abstraction and

Feature Propagation layers is followed by ReLU and batch-norm layers. Our network architecture is:

$$SA\left(0.2, 0.1, [F, 64, 128]\right) - SA\left(0.25, 0.2, [128, 128, 256]\right) -$$
$$SA\left([256, 512, 1024]\right) - FP(1, [1024 + 256, 256, 256]) -$$
$$FP(3, [256 + 128, 256, 128]) - FP(3, [128 + F, 128, 128]) -$$
$$FC(128, 128) - FC(128, 10)$$

where F is the number of input features and the final layer outputs scores for the 10 contact value classes.

B.1.2 Image Encoder-Decoder

We take inspiration from U-Net [96] and design the light-weight network shown in B.1 that extracts dense features from RGB images. The global average pooling layer is intended to capture information about the entire hand and object.

B.2 Training and Evaluation Details

All models are trained using PyTorch [104] and the Adam optimizer [157] (base learning rate $\in \{5 \times 10^{-4}, 1 \times 10^{-3}, 5 \times 10^{-3}\}$, momentum of 0.9, weight decay of $5e-4$, and a batch size of 25). Both point-clouds and voxel-grids are rotated around their 'up'-axis at regularly spaced 30 degree intervals. These rotations are considered separate data points during training, and their predictions are averaged during evaluation.

For image-based contact prediction, ContactPose has approximately 300 RGB-D frames ($\times$ 3 Kinects) for each grasp, but temporally nearby frames are highly correlated because of the high frame rate. Hence, we include equally spaced 50 frames from each grasp in the training set. Evaluation is performed over equally spaced 12 frames from this set of 50 frames.

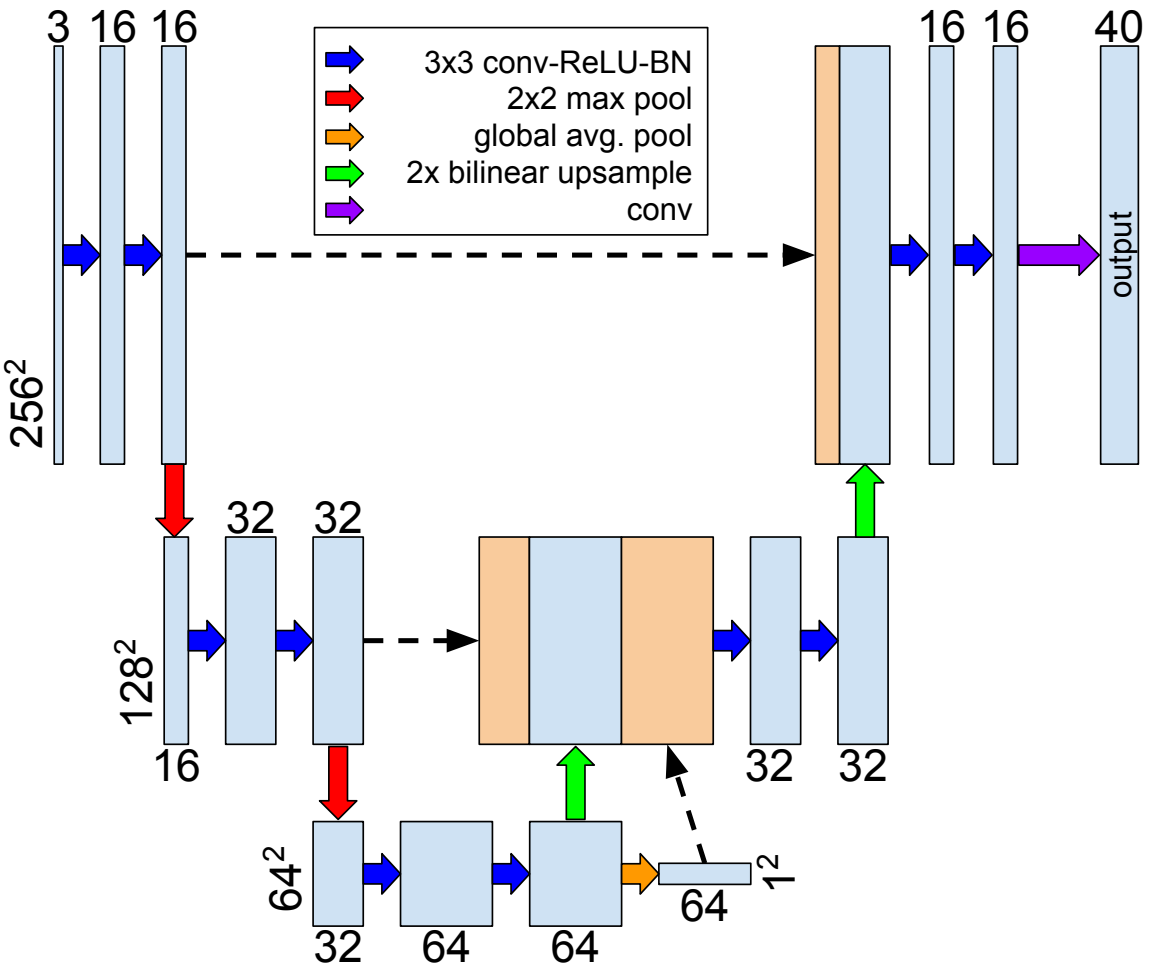


Figure B.1: Architecture for the image encoder-decoder from Figure 3.10a. Horizontal numbers indicate number of channels, and vertical numbers indicate spatial dimensions.

B.3 MANO Fitting

This section provides details for the fitting procedure of the MANO [76] hand model to ContactPose data. Borrowing notation from [76], the MANO model is a mesh with vertices $M(\beta, \theta)$ parameterized by shape parameters β and pose parameters θ. The 3D joint locations of the posed mesh, denoted here by $J(\beta, \theta)$, are also a function of the shape and pose parameters. We modify the original model by adding one joint at each fingertip, thus matching the format of joints J^* in ContactPose annotations.

MANO fitting is performed by optimizing the following objective function, which combines L2 distance of 3D joints and shape parameter regularization:

$$\beta^*, \theta^* = \arg min_{\beta,\theta} ||J(\beta, \theta) - J^*|| + \frac{1}{\sigma}||\beta|| \tag{B.1}$$

where σ is set to 10. It is optimized using the Dogleg [158] optimizer implemented in chumpy [159]. We initialized β and θ to $\mathbf{0}$ (mean shape and pose) after 6-DOF alignment of the wrist and 5 palm joints. Finally, the MANO model includes a PCA decomposition of 45 pose parameters to 6 parameters by default. We used 10 pose components, finding empirically that more than 10 pose components provide marginal or no improvement to the mean joint error, while reducing mesh smoothness. This yields a mean 3D joint error of 8.2191 mm and area under the error threshold vs. accuracy curve of 0.8324, which is better than or comparable to the HO-3D [78] (joint error 7.7 mm, AUC 0.79), FreiHand [79] (AUC 0.791), and Hands 2019 [160] (joint error 11.39 mm) datasets.

B.4 Participants' Hand Information

We captured information about each ContactPose participant's hands in two ways: 1) contact map on a flat plate (example shown in Figure B.2), and 2) RGB-D videos of the participants performing 7 hand gestures (shown in Figure B.3). This can

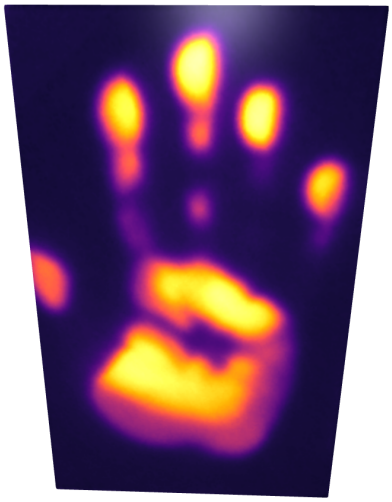


Figure B.2: Contact map of a participant's palm on a flat plate. Such palm contact maps for each participant are included in ContactPose.

potentially be used to estimate the hand shape by fit embodied hand models (e.g. [76]).

B.5 List of Objects

Table B.1 shows a list of all 25 objects in ContactPose, along with information about the which of these objects are included in the two functional grasping categories, and the specific 'use' instructions.

B.6 Example Data from ContactPose

RGB-D Images with Projected Hand Pose: Figure B.4 shows example RGB and depth images (256×256 crops centered around the object) for all objects, along with projected 3D joints.

Hand Contact Probabilities: Figure B.5 shows (phalange-level) hand-part contact probabilities (similar to Figure 3.5a) for all objects, averaged separately over 'use' and 'hand-off' grasps. Many objects that elicit significantly different 'use' and 'hand-off' contact patterns, e.g. cellphone, flashlight, hammer, knife, mouse, pan, PS controller, stapler, toothbrush, and toothpaste. The 'use' grasps for banana and water-bottle

Figure B.3: Pre-defined hand gestures performed by each participant. RGB-D videos from 3 Kinects of each participant performing these gestures are included in Contact-Pose.

Table B.1: List of objects in ContactPose and specific 'use' instructions

Object	**handoff**	**use**	**use instruction**
apple	✓	✓	eat
banana	✓	✓	peel
binoculars	✓	✓	see through
bowl	✓	✓	drink from
camera	✓	✓	take picture
cell phone	✓	✓	talk on
cup	✓	✓	drink from
door knob		✓	twist to open door
eyeglasses	✓	✓	wear
flashlight	✓	✓	turn on
hammer	✓	✓	hit a nail
headphones	✓	✓	wear
knife	✓	✓	cut
light bulb	✓	✓	screw in a socket
mouse	✓	✓	use to point and click
mug	✓	✓	drink from
pan	✓	✓	cook in
PS controller	✓	✓	play a game with
scissors	✓	✓	cut with
stapler	✓	✓	staple
toothbrush	✓	✓	brush teeth
toothpaste	✓	✓	squeeze out toothpaste
Utah teapot	✓	✓	pour tea from
water bottle	✓	✓	open
wine glass	✓	✓	drink wine from
Total	**24**	**25**	

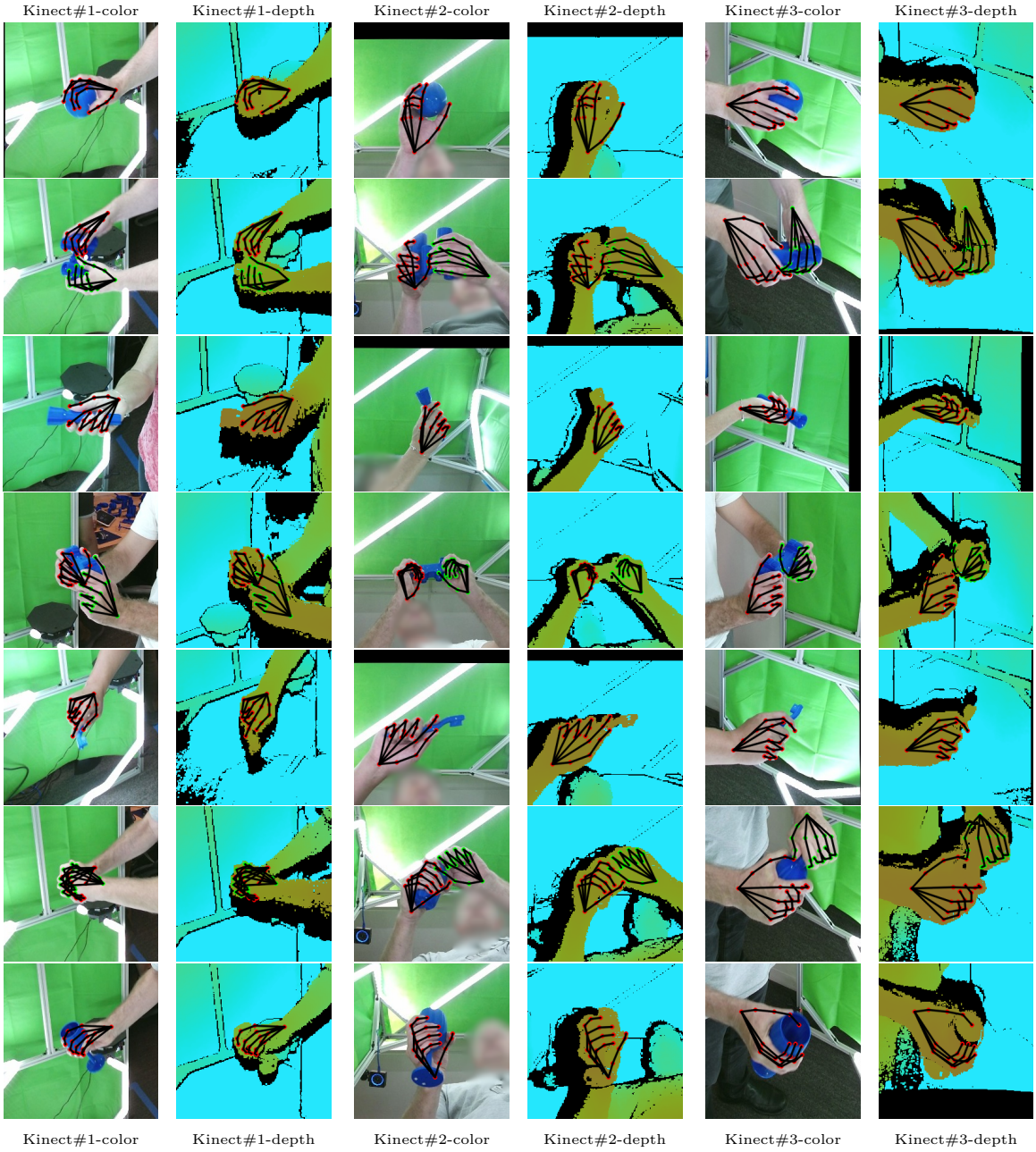


Figure B.4: Example RGB and depth images from ContactPose ('use' intention), with 3D joint locations projected into the images. Left hand joints are **green**, right hand joints are **red**.

have different contact patterns on the left and right hand, because many participants use their non-dominant hand to hold them firmly in an enveloping grasp and the dominant hand to peel and open the cap, respectively.

Grasps: To further demonstrate the scale and diversity of ContactPose data, we present a slice of the data. Figures B.5 and B.5 show all the 'use' and 'hand-off' grasps (contact map and hand pose) for one object (PS-controller), respectively. Note the significant influence of intent on grasps, and also the intra-intent diversity of grasps.

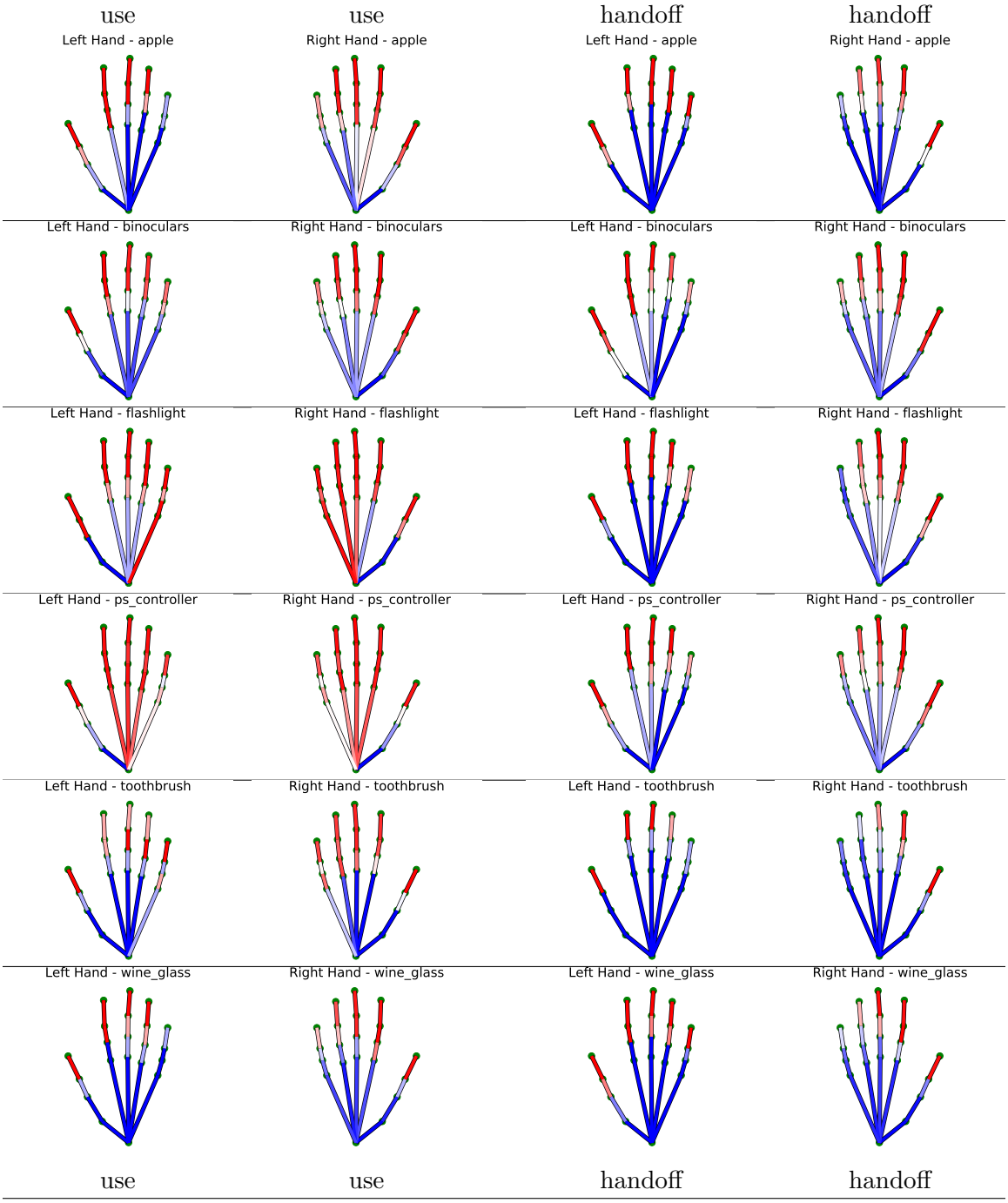


Figure B.5: Hand-part contact probabilities for objects in ContactPose (**red** indicates high probability and **blue** indicates low probability).

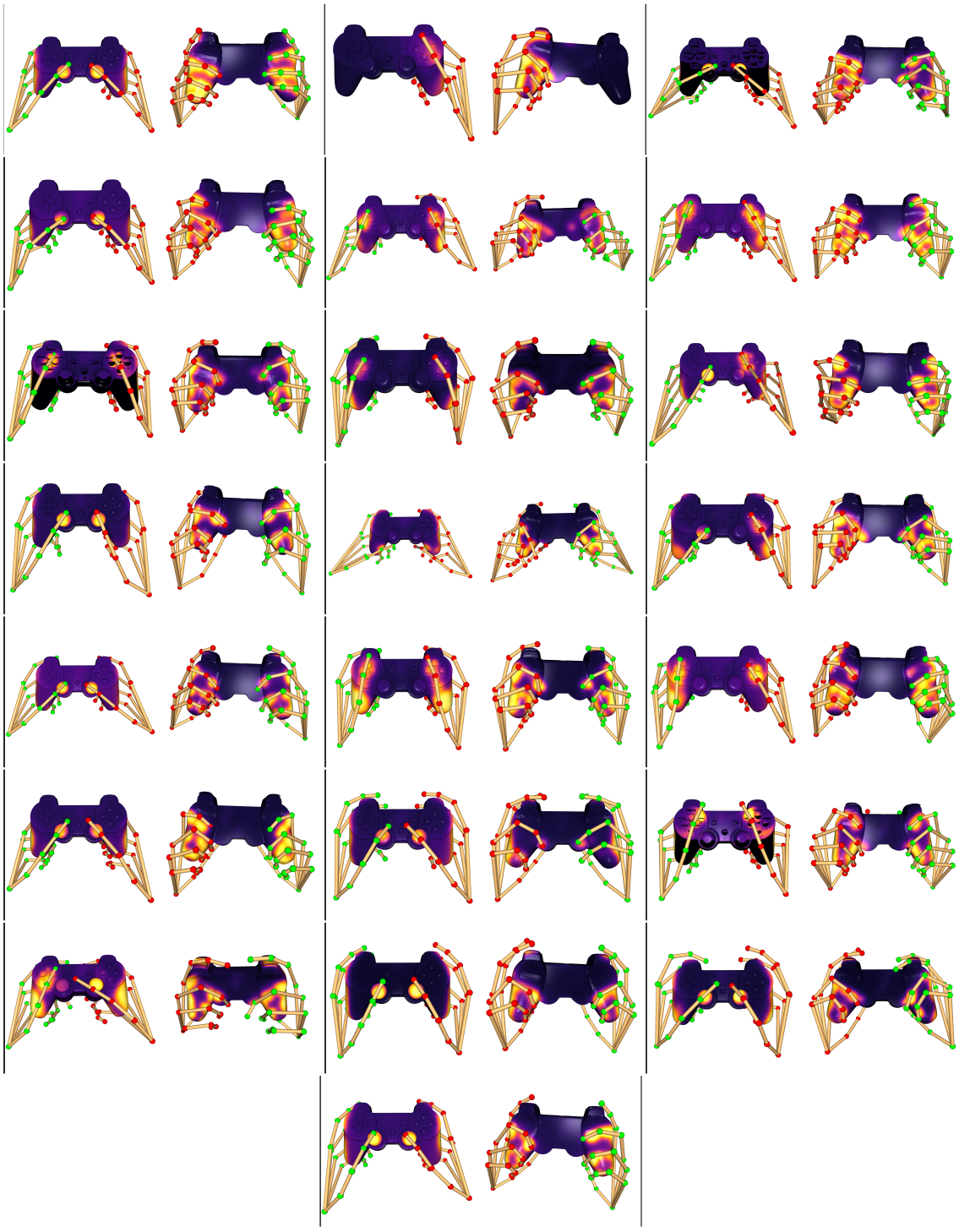


Figure B.5: A slice through ContactPose: Some PS-controller ‘use’ grasps (2 views per grasp).

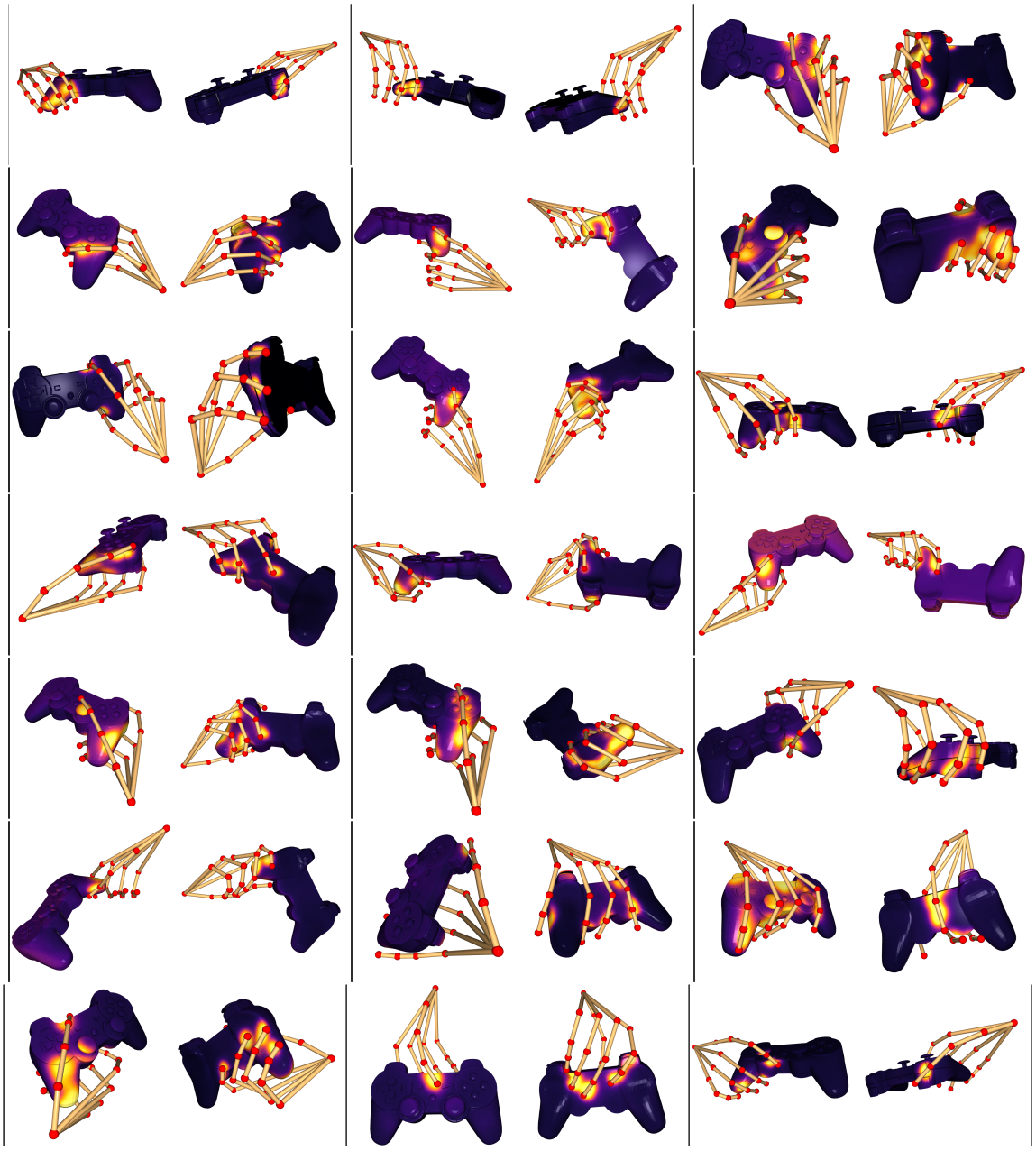


Figure B.5: A slice through ContactPose: Some PS-controller 'hand-off' grasps (2 views per grasp).

REFERENCES

[1] U. Castiello, "The neuroscience of grasping", *Nature Reviews Neuroscience*, vol. 6, no. 9, p. 726, 2005.

[2] C. Ansuini, L. Giosa, L. Turella, G. Altoè, and U. Castiello, "An object for an action, the same object for other actions: Effects on hand shaping", *Experimental Brain Research*, vol. 185, no. 1, pp. 111–119, 2008.

[3] L. Sartori, E. Straulino, and U. Castiello, "How objects are grasped: The interplay between affordances and end-goals", *PloS one*, vol. 6, no. 9, e25203, 2011.

[4] G. Heumer, H. B. Amor, M. Weber, and B. Jung, "Grasp recognition with uncalibrated data gloves-a comparison of classification methods", in *Virtual Reality Conference, 2007. VR'07. IEEE*, IEEE, 2007, pp. 19–26.

[5] Y. Lin and Y. Sun, "Grasp planning based on strategy extracted from demonstration", in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, IEEE, 2014, pp. 4458–4463.

[6] Y. C. Nakamura, D. M. Troniak, A. Rodriguez, M. T. Mason, and N. S. Pollard, "The complexities of grasping in the wild", in *Humanoid Robotics (Humanoids), 2017 IEEE-RAS 17th International Conference on*, IEEE, 2017, pp. 233–240.

[7] A. Saudabayev, Z. Rysbek, R. Khassenova, and H. A. Varol, "Human grasping database for activities of daily living with depth, color and kinematic data streams", *Scientific data*, vol. 5, 2018.

[8] D. Tang, H. Jin Chang, A. Tejani, and T.-K. Kim, "Latent regression forest: Structured estimation of 3d articulated hand posture", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 3786–3793.

[9] R. Balasubramanian, L. Xu, P. D. Brook, J. R. Smith, and Y. Matsuoka, "Physical human interactive guidance: Identifying grasping principles from human-planned grasps", *IEEE Transactions on Robotics*, vol. 4, no. 28, pp. 899–910, 2012.

[10] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks", *ACM Transactions on Graphics (ToG)*, vol. 33, no. 5, p. 169, 2014.

[11] Y. Yang, C. Fermuller, Y. Li, and Y. Aloimonos, "Grasp type revisited: A modern perspective on a classical feature for vision", in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015.

[12] D.-A. Huang, M. Ma, W.-C. Ma, and K. M. Kitani, "How do we use our hands? discovering a diverse set of common grasps", in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015.

[13] I. M. Bullock, T. Feix, and A. M. Dollar, "The yale human grasping dataset: Grasp, object, and task data in household and machine shop environments", *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 251–255, 2015.

[14] S. Yuan, Q. Ye, B. Stenger, S. Jain, and T.-K. Kim, "Bighand2. 2m benchmark: Hand pose dataset and state of the art analysis", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4866–4874.

[15] G. Garcia-Hernando, S. Yuan, S. Baek, and T.-K. Kim, "First-person hand action benchmark with rgb-d videos and 3d hand pose annotations", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 409–419.

[16] M. R. Cutkosky, "On grasp choice, grasp models, and the design of hands for manufacturing tasks", *IEEE Transactions on robotics and automation*, vol. 5, no. 3, pp. 269–279, 1989.

[17] N. Kamakura, M. Matsuo, H. Ishii, F. Mitsuboshi, and Y. Miura, "Patterns of static prehension in normal hands", *American Journal of Occupational Therapy*, vol. 34, no. 7, pp. 437–445, 1980.

[18] K. Bernardin, K. Ogawara, K. Ikeuchi, and R. Dillmann, "A sensor fusion approach for recognizing continuous human grasping sequences using hidden markov models", *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 47–57, 2005.

[19] T.-H. Pham, N. Kyriazis, A. A. Argyros, and A. Kheddar, "Hand-object contact force estimation from markerless visual tracking", *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2883–2896, 2018.

[20] S. Puhlmann, F. Heinemann, O. Brock, and M. Maertens, "A compact representation of human single-object grasping", in *2016 IEEE International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2016, pp. 1954–1959.

[21] G. Rogez, J. S. Supancic, and D. Ramanan, "Understanding everyday hands in action from rgb-d images", in *2015 IEEE International Conference on Computer Vision (ICCV)*, IEEE, 2015, pp. 3889–3897.

[22] M. Lau, K. Dev, W. Shi, J. Dorsey, and H. Rushmeier, "Tactile mesh saliency", *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, p. 52, 2016.

[23] S. Akizuki and Y. Aoki, "Tactile logging for understanding plausible tool use based on human demonstration", in *British Machine Vision Conference 2018, BMVC 2018, Northumbria University, Newcastle, UK, September 3-6, 2018*, 2018, p. 334.

[24] H. Hamer, J. Gall, T. Weise, and L. Van Gool, "An object-dependent hand pose prior from sparse training data", in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, 2010, pp. 671–678.

[25] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics", *arXiv preprint arXiv:1703.09312*, 2017.

[26] C. Choi, W. Schwarting, J. DelPreto, and D. Rus, "Learning object grasping for soft robot hands", *IEEE Robotics and Automation Letters*, 2018.

[27] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps", *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[28] S. Lee, S. P. S. Prakash, M. Cogswell, V. Ranjan, D. Crandall, and D. Batra, "Stochastic multiple choice learning for training diverse deep ensembles", in *Advances in Neural Information Processing Systems*, 2016, pp. 2119–2127.

[29] M. Firman, N. D. Campbell, L. Agapito, and G. J. Brostow, "Diversenet: When one right answer is not enough", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5598–5607.

[30] G. Ghazaei, I. Laina, C. Rupprecht, F. Tombari, N. Navab, and K. Nazarpour, "Dealing with ambiguity in robotic grasping via multiple predictions", *arXiv preprint arXiv:1811.00793*, 2018.

[31] R. Luo, O. Sener, and S. Savarese, "Scene semantic reconstruction from egocentric rgb-d-thermal videos", in *2017 International Conference on 3D Vision (3DV)*, IEEE, 2017, pp. 593–602.

[32] E. Larson, G. Cohn, S. Gupta, X. Ren, B. Harrison, D. Fox, and S. Patel, "Heatwave: Thermal imaging for surface user interaction", in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '11, Vancouver, BC, Canada: ACM, 2011, pp. 2565–2574, ISBN: 978-1-4503-0228-9.

[33] M. Vollmer and K.-P. Möllmann, *Infrared thermal imaging: fundamentals, research and applications*. John Wiley & Sons, 2017.

[34] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols", *arXiv preprint arXiv:1502.03143*, 2015.

[35] N. Correll, K. E. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. M. Romano, and P. R. Wurman, "Analysis and observations from the first amazon picking challenge", *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 1, pp. 172–188, 2018.

[36] R. H. Cuijpers, J. B. Smeets, and E. Brenner, "On the relation between object shape and grasping kinematics", *Journal of Neurophysiology*, vol. 91, no. 6, pp. 2598–2606, 2004.

[37] M. Quigley, J. Faust, T. Foote, and J. Leibs, "Ros: An open-source robot operating system".

[38] P. Besl and N. D. McKay, "A method for registration of 3-d shapes", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 14, no. 2, pp. 239–256, 1992.

[39] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)", in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 2011.

[40] Q.-Y. Zhou and V. Koltun, "Color map optimization for 3d reconstruction with consumer depth cameras", *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, p. 155, 2014.

[41] Q.-Y. Zhou, J. Park, and V. Koltun, "Open3D: A modern library for 3D data processing", *arXiv:1801.09847*, 2018.

[42] L. Kaufman and P. Rousseeuw, *Clustering by means of medoids*. North-Holland, 1987.

[43] T. Schmidt, K. Hertkorn, R. Newcombe, Z. Marton, M. Suppa, and D. Fox, "Depth-based tracking with physical constraints for robot manipulation", in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2015, pp. 119–126.

[44] Y. Ye and C. K. Liu, "Synthesis of detailed hand manipulations using contact sampling", *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, p. 41, 2012.

[45] R. Deimel and O. Brock, "A novel type of compliant and underactuated robotic hand for dexterous grasping", *The International Journal of Robotics Research*, vol. 35, no. 1-3, pp. 161–185, 2016.

[46] K. C. Galloway, K. P. Becker, B. Phillips, J. Kirby, S. Licht, D. Tchernov, R. J. Wood, and D. F. Gruber, "Soft robotic grippers for biological sampling on deep reefs", *Soft robotics*, vol. 3, no. 1, pp. 23–33, 2016.

[47] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks", in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2017, pp. 5967–5976.

[48] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks", in *IEEE International Conference on Computer Vision*, 2017.

[49] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks", in *Advances in Neural Information Processing Systems*, 2017, pp. 700–708.

[50] M. Mirza and S. Osindero, "Conditional generative adversarial nets", *CoRR*, vol. abs/1411.1784, 2014. arXiv: `1411.1784`.

[51] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets", in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[52] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 652–660.

[53] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition", in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, IEEE, 2015, pp. 922–928.

[54] C. Rupprecht, I. Laina, R. DiPietro, M. Baust, F. Tombari, N. Navab, and G. D. Hager, "Learning in an uncertain world: Representing ambiguity through multiple hypotheses", in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3591–3600.

[55] F. S. Nooruddin and G. Turk, "Simplification and repair of polygonal models using volumetric techniques", *IEEE Transactions on Visualization and Computer Graphics*, vol. 9, no. 2, pp. 191–205, 2003.

[56] P. Min, *Binvox*, `http://www.patrickmin.com/binvox`, Accessed: 2018-11-16, 2004 - 2017.

[57] S. Sridhar, F. Mueller, M. Zollhöfer, D. Casas, A. Oulasvirta, and C. Theobalt, "Real-time joint tracking of a hand manipulating an object from rgb-d input", in *European Conference on Computer Vision*, Springer, 2016, pp. 294–310.

[58] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks", *ACM Transactions on Graphics (ToG)*, vol. 33, no. 5, p. 169, 2014.

[59] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping", in *CVPR*, 2017.

[60] G. Garcia-Hernando, S. Yuan, S. Baek, and T.-K. Kim, "First-person hand action benchmark with rgb-d videos and 3d hand pose annotations", in *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, 2018.

[61] B. Tekin, F. Bogo, and M. Pollefeys, "H+ o: Unified egocentric recognition of 3d hand-object poses and interactions", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4511–4520.

[62] D. Tzionas, L. Ballan, A. Srikantha, P. Aponte, M. Pollefeys, and J. Gall, "Capturing hands in action using discriminative salient points and physics simulation", *International Journal of Computer Vision*, vol. 118, no. 2, pp. 172–193, 2016.

[63] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid, "Learning joint reconstruction of hands and manipulated objects", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 807–11 816.

[64] M. Hassan, V. Choutas, D. Tzionas, and M. J. Black, "Resolving 3d human pose ambiguities with 3d scene constraints", in *The IEEE International Conference on Computer Vision (ICCV)*, Oct. 2019.

[65] X. Zhang, Q. Li, H. Mo, W. Zhang, and W. Zheng, "End-to-end hand mesh recovery from a monocular rgb image", in *The IEEE International Conference on Computer Vision (ICCV)*, Oct. 2019.

[66] S. Brahmbhatt, A. Handa, J. Hays, and D. Fox, "ContactGrasp: Functional Multi-finger Grasp Synthesis from Contact", in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[67] Q. Lu, K. Chenna, B. Sundaralingam, and T. Hermans, "Planning multi-fingered grasps as probabilistic inference in a learned deep network", in *International Symposium on Robotics Research*, 2017.

[68] C. Ferrari and J. Canny, "Planning optimal grasps", in *Proceedings 1992 IEEE International Conference on Robotics and Automation*, IEEE, pp. 2290–2295.

[69] N. S. Pollard, "Parallel methods for synthesizing whole-hand grasps from generalized prototypes", MASSACHUSETTS INST OF TECH CAMBRIDGE ARTIFICIAL INTELLIGENCE LAB, Tech. Rep., 1994.

[70] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping", *IEEE Robotics & Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.

[71] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg, "Learning ambidextrous robot grasping policies", *Science Robotics*, vol. 4, no. 26, eaau4984, 2019.

[72] R. Deimel and O. Brock, "A novel type of compliant and underactuated robotic hand for dexterous grasping", *The International Journal of Robotics Research*, vol. 35, no. 1-3, pp. 161–185, 2016.

[73] B. S. Homberg, R. K. Katzschmann, M. R. Dogar, and D. Rus, "Haptic identification of objects using a modular soft robotic gripper", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2015, pp. 1698–1705.

[74] J. Wade, T. Bhattacharjee, R. D. Williams, and C. C. Kemp, "A force and thermal sensing skin for robots in human environments", *Robotics and Autonomous Systems*, vol. 96, pp. 1–14, 2017.

[75] S. Sundaram, P. Kellnhofer, Y. Li, J.-Y. Zhu, A. Torralba, and W. Matusik, "Learning the signatures of the human grasp using a scalable tactile glove", *Nature*, vol. 569, no. 7758, p. 698, 2019.

[76] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together", *ACM Transactions on Graphics (TOG)*, vol. 36, no. 6, p. 245, 2017.

[77] S. Brahmbhatt, C. Ham, C. C. Kemp, and J. Hays, "ContactDB: Analyzing and predicting grasp contact via thermal imaging", in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019.

[78] S. Hampali, M. Oberweger, M. Rad, and V. Lepetit, "HO-3D: A multi-user, multi-object dataset for joint 3d hand-object pose estimation", *arXiv preprint arXiv:1907.01481*, 2019.

[79] C. Zimmermann, D. Ceylan, J. Yang, B. Russell, M. Argus, and T. Brox, "Freihand: A dataset for markerless capture of hand pose and shape from single rgb images", in *The IEEE International Conference on Computer Vision (ICCV)*, Oct. 2019.

[80] T.-H. Pham, A. Kheddar, A. Qammaz, and A. A. Argyros, "Towards force sensing from vision: Observing hand-object interactions to infer manipulation forces", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2810–2819.

[81] T.-H. Pham, N. Kyriazis, A. A. Argyros, and A. Kheddar, "Hand-object contact force estimation from markerless visual tracking", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, pp. 2883–2896, 2018.

[82] Y. Ye and C. K. Liu, "Synthesis of detailed hand manipulations using contact sampling", *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, p. 41, 2012.

[83] J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, and K. Goldberg, "Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards", in *2016 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2016, pp. 1957–1964.

[84] M. Teschner, S. Kimmerle, B. Heidelberger, G. Zachmann, L. Raghupathi, A. Fuhrmann, M.-P. Cani, F. Faure, N. Magnenat-Thalmann, W. Strasser, *et al.*, "Collision detection for deformable objects", in *Computer graphics forum*, Wiley Online Library, vol. 24, 2005, pp. 61–81.

[85] E. Larsen, S. Gottschalk, M. C. Lin, and D. Manocha, "Fast distance queries with rectangular swept sphere volumes", in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, IEEE, vol. 4, 2000, pp. 3719–3726.

[86] L. Ballan, A. Taneja, J. Gall, L. Van Gool, and M. Pollefeys, "Motion capture of hands in action using discriminative salient points", in *European Conference on Computer Vision*, Springer, 2012, pp. 640–653.

[87] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3d hands, face, and body from a single image", in *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019.

[88] T. Feix, J. Romero, H.-B. Schmiedmayer, A. M. Dollar, and D. Kragic, "The grasp taxonomy of human grasp types", *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 1, pp. 66–77, 2015.

[89] I. M. Bullock, T. Feix, and A. M. Dollar, "The yale human grasping dataset: Grasp, object, and task data in household and machine shop environments", *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 251–255, 2015.

[90] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN", in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 2980–2988.

[91] P. J. Huber, "Robust estimation of a location parameter", in *Breakthroughs in statistics*, Springer, 1992, pp. 492–518.

[92] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography", *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[93] I. M. Bullock, J. Z. Zheng, S. De La Rosa, C. Guertler, and A. M. Dollar, "Grasp frequency and usage in daily household and machine shop tasks", *IEEE transactions on haptics*, vol. 6, no. 3, pp. 296–308, 2013.

[94] *BioTac*, `https://www.syntouchinc.com/robotics/`, Accessed: 2020-03-05.

[95] R. J. G. B. Campello, D. Moulavi, A. Zimek, and J. Sander, "Hierarchical density estimates for data clustering, visualization, and outlier detection", *ACM Trans. Knowl. Discov. Data*, vol. 10, no. 1, 5:1–5:51, Jul. 2015.

[96] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation", in *International Conference on Medical image computing and computer-assisted intervention*, Springer, 2015, pp. 234–241.

[97] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context", in *European conference on computer vision*, Springer, 2014, pp. 740–755.

[98] X. Zhou, S. Leonardos, X. Hu, and K. Daniilidis, "3d shape estimation from 2d landmarks: A convex relaxation approach", in *proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4447–4455.

[99] T. Groueix, M. Fisher, V. G. Kim, B. C. Russell, and M. Aubry, "A papier-mâché approach to learning 3d surface generation", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 216–224.

[100] M. Garon and J.-F. Lalonde, "Deep 6-dof tracking", *IEEE transactions on visualization and computer graphics*, vol. 23, no. 11, pp. 2410–2418, 2017.

[101] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield, "Deep object pose estimation for semantic robotic grasping of household objects", *arXiv preprint arXiv:1809.10790*, 2018.

[102] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization", in *European conference on computer vision*, Springer, 2016, pp. 649–666.

[103] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space", in *Advances in neural information processing systems*, 2017, pp. 5099–5108.

[104] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch", in *NIPS Autodiff Workshop*, 2017.

[105] M. Fey and J. E. Lenssen, "Fast graph representation learning with PyTorch Geometric", in *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[106] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition", in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2015, pp. 922–928.

[107] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification", in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[108] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", in *International Conference on Machine Learning*, 2015, pp. 448–456.

[109] H. Joo, T. Simon, and Y. Sheikh, "Total capture: A 3d deformation model for tracking faces, hands, and bodies", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8320–8329.

[110] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics", 2017.

[111] J. Mahler, M. Matl, X. Liu, A. Li, D. Gealy, and K. Goldberg, "Dex-net 3.0: Computing robust robot suction grasp targets in point clouds using a new analytic model and deep learning", *arXiv preprint arXiv:1709.06670*, 2017.

[112] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours", in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2016, pp. 3406–3413.

[113] M Ciocarlie, C Goldfeder, and P Allen, "Dimensionality reduction for hand-independent dexterous robotic grasping", in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*.

[114] M. A. Roa, M. J. Argus, D. Leidner, C. Borst, and G. Hirzinger, "Power grasp planning for anthropomorphic robot hands", in *2012 IEEE International Conference on Robotics and Automation*, IEEE, 2012, pp. 563–569.

[115] M. Przybylski, T. Asfour, and R. Dillmann, "Planning grasps for robotic hands using a novel object representation based on the medial axis transform", in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2011, pp. 1781–1788.

[116] S. Ekvall and D. Kragic, "Learning and evaluation of the approach vector for automatic grasp generation and planning", in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, IEEE, 2007, pp. 4715–4720.

[117] J. Romero, H. Kjellstrom, and D. Kragic, "Modeling and evaluation of human-to-robot mapping of grasps", in *2009 International Conference on Advanced Robotics*, IEEE, 2009, pp. 1–6.

[118] A. Herzog, P. Pastor, M. Kalakrishnan, L. Righetti, T. Asfour, and S. Schaal, "Template-based learning of grasp selection", in *2012 IEEE International Conference on Robotics and Automation*, IEEE, 2012, pp. 2379–2384.

[119] G. Garcia-Hernando, S. Yuan, S. Baek, and T.-K. Kim, "First-person hand action benchmark with rgb-d videos and 3d hand pose annotations", in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018.

[120] A. Bicchi and V. Kumar, "Robotic grasping and contact: A review", in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, IEEE, vol. 1, 2000, pp. 348–353.

[121] D. Prattichizzo, M. Malvezzi, M. Gabiccini, and A. Bicchi, "On the manipulability ellipsoids of underactuated robotic hands with compliance", *Robotics and Autonomous Systems*, vol. 60, no. 3, pp. 337–346, 2012.

[122] C. Rosales, R. Suárez, M. Gabiccini, and A. Bicchi, "On the synthesis of feasible and prehensile robotic grasps", in *2012 IEEE International Conference on Robotics and Automation*, IEEE, 2012, pp. 550–556.

[123] V.-D. Nguyen, "Constructing force-closure grasps", *The International Journal of Robotics Research*, vol. 7, no. 3, pp. 3–16, 1988.

[124] M. A. Roa and R. Suárez, "Computation of independent contact regions for grasping 3-d objects", *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 839–850, 2009.

[125] R. Krug, D. Dimitrov, K. Charusta, and B. Iliev, "On the efficient computation of independent contact regions for force closure grasps", in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2010, pp. 586–591.

[126] A. Rodriguez, M. T. Mason, and S. Ferry, "From caging to grasping", *The International Journal of Robotics Research*, vol. 31, no. 7, pp. 886–900, 2012.

[127] J. Seo, S. Kim, and V. Kumar, "Planar, bimanual, whole-arm grasping", in *2012 IEEE International Conference on Robotics and Automation*, IEEE, 2012, pp. 3271–3277.

[128] Y. Li and N. Pollard, "A shape matching algorithm for synthesizing humanlike enveloping grasps", in *5th IEEE-RAS International Conference on Humanoid Robots, 2005.*, IEEE, 2005, pp. 442–449.

[129] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps", *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[130] Y. Jiang, S. Moseson, and A. Saxena, "Efficient grasping from rgbd images: Learning using a new rectangle representation", in *2011 IEEE International Conference on Robotics and Automation*, IEEE, 2011, pp. 3304–3311.

[131] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks", *CoRR*, vol. abs/1412.3128, 2014. arXiv: 1412.3128.

[132] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesis—a survey", *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289–309, 2014.

[133] A. T. Miller and P. K. Allen, "Examples of 3d grasp quality computations", in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, IEEE, vol. 2, 1999, pp. 1240–1246.

[134] ——, "Graspit! a versatile simulator for robotic grasping", *IEEE Robotics & Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.

[135] D. Song, C. H. Ek, K. Huebner, and D. Kragic, "Multivariate discretization for bayesian network structure learning in robot grasping", in *2011 IEEE International Conference on Robotics and Automation*, IEEE, 2011, pp. 1944–1950.

[136] H. B. Amor, O. Kroemer, U. Hillenbrand, G. Neumann, and J. Peters, "Generalization of human grasping for multi-fingered robot hands", in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2012, pp. 2043–2050.

[137] J. Varley, J. Weisz, J. Weiss, and P. Allen, "Generating multi-fingered robotic grasps via deep learning", in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2015, pp. 4415–4420.

[138] Y. Ye and C. K. Liu, "Synthesis of detailed hand manipulations using contact sampling", *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, p. 41, 2012.

[139] T. Schmidt, R. Newcombe, and D. Fox, "DART: Dense Articulated Real-Time Tracking", in *Proceedings of Robotics: Science and Systems*, Berkeley, USA, Jul. 2014.

[140] SimLab. (). "Allegro Hand", (visited on 02/19/2019).

[141] B. Tech. (). "Barrett Hand", (visited on 02/19/2019).

[142] T. Schmidt, K. Hertkorn, R. Newcombe, Z. Marton, M. Suppa, and D. Fox, "Depth-based tracking with physical constraints for robot manipulation", in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, IEEE, 2015, pp. 119–126.

[143] M. R. Cutkosky and R. D. Howe, "Human grasp choice and robotic grasp analysis", in *Dextrous robot hands*, Springer, 1990, pp. 5–31.

[144] R. Pelossof, A. Miller, P. Allen, and T. Jebara, "An svm learning approach to robotic grasping", in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, IEEE, vol. 4, 2004, pp. 3512–3518.

[145] T. Tosun, R. Mead, and R. Stengel, "A general method for kinematic retargeting: Adapting poses between humans and robots", in *ASME 2014 International Mechanical Engineering Congress and Exposition*, American Society of Mechanical Engineers, 2014, V04AT04A027–V04AT04A027.

[146] M. R. Cutkosky, "On grasp choice, grasp models, and the design of hands for manufacturing tasks", *IEEE Transactions on robotics and automation*, vol. 5, no. 3, pp. 269–279, 1989.

[147] D. M. Wolpert and J. R. Flanagan, "Motor prediction", *Current biology*, vol. 11, no. 18, R729–R732, 2001.

[148] M. N. Loh, L. Kirsch, J. C. Rothwell, R. N. Lemon, and M. Davare, "Information about the weight of grasped objects from vision and internal models interacts within the primary motor cortex", *Journal of Neuroscience*, vol. 30, no. 20, pp. 6984–6990, 2010.

[149] K. Ehsani, S. Tulsiani, S. Gupta, A. Farhadi, and A. Gupta, "Use the force, luke! learning to predict physical forces by simulating effects", in *CVPR*, 2020.

[150] Y. Sun, J. M. Hollerbach, and S. A. Mascaro, "Measuring fingertip forces by imaging the fingernail", in *2006 14th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, IEEE, 2005, pp. 125–131.

[151] S. A. Mascaro and H. H. Asada, "Understanding of fingernail-bone interaction and fingertip hemodynamics for fingernail sensor design", in *Haptic Interfaces for Virtual Environment and Teleoperator Systems, International Symposium on*, 2002, pp. 113–113.

[152] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Second. Cambridge University Press, ISBN: 0521540518, 2004.

[153] Z Cao, G. M. Hidalgo, T Simon, S. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields.", *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[154] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs", *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[155] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation", in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818.

[156] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation", in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[157] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", *arXiv preprint arXiv:1412.6980*, 2014.

[158] M. J. Powell, "A new algorithm for unconstrained optimization", in *Nonlinear programming*, Elsevier, 1970, pp. 31–65.

[159] *chumpy: Autodifferentiation tool for Python*, `https://github.com/mattloper/chumpy`, Accessed: 2020-03-12.

[160] *5th International Workshop on Observing and Understanding Hands in Action*, `https://sites.google.com/view/hands2019/challenge#h.p_adfpp7VAhgAL`, Accessed: 2020-03-12.

VITA

Samarth Brahmbhatt is a computer vision, robotics, and machine learning researcher. He is currently a PhD student in the Robotics program at the Georgia Institute of Technology in Atlanta, GA, USA. His PhD thesis is supervised by Dr James Hays, and he also collaborates with Dr Charles C. Kemp. It focuses on observing, modeling, and imitating human functional grasping behavior, especially from the contact perspective.

Samarth holds a masters degree in Robotics from the University of Pennsylvania, Philadelphia, PA, USA and a bachelors degree in Electronics and Communication Engineering from Nirma University, Ahmedabad, India. He was born in Vallabh Vidyanagar, Gujarat, India and grew up in the city of Gandhinagar.